



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

“Μελέτη και σχεδιασμός συστημάτων πλοήγησης σε εξωτερικούς  
χώρους”

“Study and design of outdoor navigation systems”

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ

της

Γιάκα Χρυσούλα

Βόλος, Οκτώβριος 2017



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ  
ΥΠΟΛΟΓΙΣΤΩΝ

“Μελέτη και σχεδιασμός συστημάτων πλοήγησης σε εξωτερικούς  
χώρους”

“Study and design of outdoor navigation systems”

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

της

Γιάκα Χρυσούλα

**Επιβλέποντες :**

**ΠΡΩΤΟΣ ΕΠΙΒΛΕΠΩΝ**

**Τσομπανοπούλου Παναγιώτα**  
Αναπληρώτρια Καθηγήτρια Π.Θ.

**ΔΕΥΤΕΡΟΣ ΕΠΙΒΛΕΠΩΝ**

**Τσουκαλάς Ελευθέριος**  
Καθηγητής Π.Θ.

**ΤΡΙΤΟΣ ΕΠΙΒΛΕΠΩΝ**

**Βασιλακόπουλος Μιχαήλ**  
Αναπληρωτής Καθηγητής Π.Θ.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την *ΗΜΕΡΟΜΗΝΙΑ ΕΞΕΤΑΣΗΣ*

*(Υπογραφή)*

.....  
ΠΡΩΤΟΣ ΕΠΙΒΛΕΠΩΝ

*(Υπογραφή)*

.....  
ΔΕΥΤΕΡΟΣ ΕΠΙΒΛΕΠΩΝ

*(Υπογραφή)*

.....  
ΤΡΙΤΟΣ ΕΠΙΒΛΕΠΩΝ

Βόλος, Οκτώβριος 2017

(Υπογραφή)

.....

**ΓΙΑΚΑ ΧΡΥΣΟΥΛΑ**

Διπλωματική Εργασία για την απόκτηση Μεταπτυχιακού Διπλώματος Ειδίκευσης στην «Επιστήμη και Τεχνολογία Υπολογιστών, Τηλεπικοινωνιών και Δικτύων», στα Πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Θεσσαλίας

© 2017 – All rights reserved

## Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τη Καθηγήτρια του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών την κα Τσομπανοπούλου Παναγιώτα για την άριστη συνεργασία, για τις χρήσιμες συμβουλές, και για τον πολύτιμο χρόνο που διέθεσε για να μου προσφέρει τις γνώσεις και την εμπειρία της κατά τη διάρκεια της εκπόνηση της μεταπτυχιακής μου εργασίας.

Τέλος, ευχαριστώ θερμά την οικογένειά μου για την αμέριστη συμπαράσταση που μου παρείχε όλα αυτά τα χρόνια για την ολοκλήρωση των μεταπτυχιακών και προπτυχιακών σπουδών μου.

## Περίληψη

Οι ραγδαίοι ρυθμοί ανάπτυξης των smartphones τα τελευταία χρόνια έχουν σαν αποτέλεσμα τη δημιουργία χιλιάδων εφαρμογών, οι οποίες προσφέρουν στο χρήστη χρήσιμες υπηρεσίες για την διευκόλυνση της καθημερινότητας του. Ωστόσο αρκετές από αυτές ήταν αποκλειστικά μόνο για Android[1] ή μόνο για iOS[2] λειτουργικά συστήματα.

Σκοπός της παρούσας μεταπτυχιακής εργασίας ήταν η δημιουργία μιας υβριδικής εφαρμογής. Δηλαδή μιας εφαρμογής που θα μπορούσε να εκτελεστεί σε οποιοδήποτε λειτουργικό περιβάλλον είτε αυτό είναι Android είτε iOS ή Windows Phone[3]. Αυτό επιτεύχθηκε με τη βοήθεια των Hybrid Application Frameworks και συγκεκριμένα του Ionic Framework[4], ενός framework που βασίζεται στο εργαλείο Cordova[5], τη γλώσσα AngularJS[6] και στη βιβλιοθήκη του Ionic τα οποία δίνουν στο developer τη δυνατότητα να δημιουργήσει υβριδικές εφαρμογές. Συνεπώς με τη βοήθεια του Ionic δημιουργήθηκε η εφαρμογή “VolosBus” που σκοπός της είναι η διευκόλυνση των χρηστών κατά τη μετακίνησή τους μέσα στη πόλη του Βόλου με την αστική συγκοινωνία.

Η εφαρμογή προσφέρει στο χρήστη όλες τις απαραίτητες πληροφορίες για το αστικό ΚΤΕΛ Βόλου[7] (δρομολόγια, στάσεις, ώρες αναχωρήσεως από την αφετηρία, εκδοτήρια εισιτηρίων), υπολογίζεται επίσης η οικονομικότερη και η βέλτιστη διαδρομή με βάση τον προορισμό του χρήστη όπως επίσης και πληροφορίες για την απόσταση που πρέπει να διανύσει μέχρι την αφετηρία κάθε λεωφορείου που επιθυμεί να επιβιβαστεί. Ακόμη, ο χρήστης έχει τη δυνατότητα να ενημερώνεται ανάλογα με τη στάση που βρίσκεται σε πόση ώρα αναμένεται η άφιξη του λεωφορείου που τον ενδιαφέρει.

Μελλοντικά προτείνεται η περαιτέρω επέκταση της εφαρμογής με επιλογές που θα προσφέρουν επιπλέον δυνατότητες στο χρήστη, όπως για παράδειγμα εφόσον θέλει ο χρήστης να ταξιδέψει εκτός Βόλου, να του προτείνεται εκτός από το αστικό λεωφορείο, ποιο τραίνο ή υπεραστικό λεωφορείο πρέπει να πάρει ανάλογα με το προορισμό του καθώς επίσης και τις ώρες αναχωρήσεως. Επίσης, ο χρήστης θα έχει τη δυνατότητα να προμηθεύεται εισιτήρια on-line μέσω της εφαρμογής τόσο για το αστικό λεωφορείο όσο και για τις μεταφορές του εκτός Βόλου.





## **Abstract**

The rapid growth rate of the smartphone industry in recent years has resulted in the creation of thousands of applications, which offer useful services to the user to facilitate his daily routine. However, several of them were exclusively for Android [1] or for iOS [2] operating systems.

The purpose of this postgraduate thesis was to create a hybrid application. An application was developed that can run on any operating system, whether this is Android or iOS or Windows Phone [3]. The development was feasible thanks to Hybrid Application Frameworks more specifically with the use of the Ionic Framework [4]. Ionic is a powerful HTML5 SDK, based on Cordova [5], AngularJS [6] and Ionic helping the developer to build hybrid applications. So, with the help of Ionic, the application "VolosBus" was created. The aim of this application is to make the transportation within the city of Volos easier, using city buses.

The application "VolosBus" provides the user with all the necessary information about the Volos City Bus [7] (timetables, stops, tickets, departures, ticket offices). Moreover, the users can find the appropriate and cheapest Service Line depending on his destination. It also indicates the most economical and optimal route based on the destination of the user as well as information on the distance which the user should walk to reach the station. Also, the user can be informed on the expected time of arrival of the bus that is of interest.

In the future, it is proposed to further extend the application with options that offer additional capabilities to the user. For example, if the user wishes to travel outside of Volos, the mobile app can propose to him, which train or intercity bus to take per his destination as well as the departure times. Also, the user will be able to purchase tickets online via the application for both the city bus and the non-Volos transport.



## Περιεχόμενα

<b>1</b>	<b>Εισαγωγή.....</b>	<b>12</b>
1.1	Η αναγκαιότητα ανάπτυξης της εφαρμογής.....	12
1.2	Παρόμοιες εφαρμογές.....	13
1.3	Διάρθρωση της Μεταπτυχιακής Εργασίας.....	13
<b>2</b>	<b>Ιστορική αναδρομή των mobile applications .....</b>	<b>15</b>
2.1	Mobile εφαρμογές.....	15
2.1.1	Το Android λειτουργικό σύστημα.....	15
2.1.2	Το IOS λειτουργικό σύστημα .....	16
2.1.3	Το Windows phone λειτουργικό σύστημα .....	17
2.2	Native Mobile εφαρμογές.....	18
<b>3</b>	<b>Hybrid Mobile Development.....</b>	<b>19</b>
3.1	Τι είναι το Hybrid Mobile Application .....	19
3.2	Hybrid Mobile Frameworks.....	21
3.2.1	Ionic .....	21
3.2.2	Sencha Touch.....	22
3.2.3	Onsen UI .....	22
3.2.4	Kendo UI .....	22
3.2.5	jQuery Mobile.....	23
3.2.6	Adobe PhoneGap.....	23
3.3	Λόγοι επιλογής του Ionic.....	24
<b>4</b>	<b>Σχεδιασμός Εφαρμογής.....</b>	<b>26</b>
4.1	Σκοπός και ανάγκες κάλυψης της εφαρμογής.....	26
4.2	Σχεδιασμός λειτουργιών εφαρμογής.....	28
4.3	Διάγραμμα ροής .....	35
<b>5</b>	<b>Υλοποίηση της εφαρμογής.....</b>	<b>37</b>

<b>5.1</b>	<b>Λεπτομέρειες υλοποίησης .....</b>	<b>37</b>
<b>5.2</b>	<b>Προγραμματιστικό περιβάλλον του <i>ionic</i> .....</b>	<b>38</b>
<b>5.3</b>	<b>Αναλυτική Παρουσίαση Αλγορίθμων .....</b>	<b>39</b>
5.3.1	Λειτουργία πλησιέστερες στάσεις.....	40
5.3.2	Αλγόριθμος Βέλτιστης διαδρομής.....	41
5.3.3	Αλγόριθμος Αφίξεων σε στάση .....	42
5.3.4	Πορεία γραμμών πάνω στο χάρτη .....	43
<b>6</b>	<b><i>Επίδειξη της εφαρμογής.....</i></b>	<b>44</b>
6.1	Γενικά.....	44
6.2	Λειτουργία Εκτέλεσης .....	45
<b>7</b>	<b><i>Συμπεράσματα .....</i></b>	<b>54</b>
7.1	Αποτελέσματα .....	54
7.2	Δυσκολίες.....	54
7.3	Μελλοντικές επεκτάσεις .....	55
<b>8</b>	<b><i>Βιβλιογραφία .....</i></b>	<b>56</b>

# ***1 Εισαγωγή***

Στο κεφάλαιο αυτό παρουσιάζονται οι λόγοι για το πώς δημιουργήθηκε η ιδέα ανάπτυξης μιας εφαρμογής που είχε σα στόχο την εξυπηρέτηση και τη βελτίωση της καθημερινότητας των κατοίκων και επισκεπτών της πόλης του Βόλου κατά τη μετακίνησή τους. Επίσης παρουσιάζονται άλλες παρόμοιες εφαρμογές που έχουν δημιουργηθεί στο παρελθόν και τέλος αφιερώνεται μια παράγραφος στο πώς έχει διαρθρωθεί η εργασία.

## ***1.1 Η αναγκαιότητα ανάπτυξης της εφαρμογής***

Ο Βόλος είναι μια όμορφη πόλη με αρχαία και πλούσια ιστορία, με γραφικά χωριά και μαγευτικές παραλίες, με αποτέλεσμα κάθε χρόνο να φιλοξενεί χιλιάδες τουρίστες. Επίσης, στον Βόλο βρίσκεται η έδρα του Πανεπιστημίου Θεσσαλίας έτσι κάθε χρόνο εκατοντάδες νέοι έρχονται να εγκατασταθούν στην περιοχή. Συνεπώς η πόλη καθημερινά είναι γεμάτη κόσμο και η μετακίνηση των ανθρώπων με τα μέσα μαζικής συγκοινωνίας είναι αναγκαία και αναπόφευκτη. Είναι γεγονός ότι κάθε μέρα χιλιάδες κόσμος κάθε ηλικίας χρησιμοποιεί τα μέσα μαζικής μεταφοράς για να εξυπηρετηθεί.

Στη σημερινή εποχή που όλα τρέχουν με πολύ γρήγορους ρυθμούς κανείς δεν θέλει να πηγαίνει σε μια στάση και να περιμένει αναμένοντας κάποιο λεωφορείο. Είναι λοιπόν χρήσιμο να γνωρίζει τι ώρα περνάει η γραμμή που τον ενδιαφέρει έτσι ώστε να κανονίζει το πρόγραμμα του και να μεταβαίνει στη συγκεκριμένη στάση ανάλογα. Επίσης, πολλοί νέοι επισκέπτες στην πόλη έχουν άγνοια σε ποια γραμμή πρέπει να επιβιβαστούν, αν θα πρέπει να πάρουν ένα ή περισσότερα λεωφορεία, σε ποιες στάσεις πρέπει να μεταβούν και πού να αποβιβαστούν ανάλογα με το προορισμό που θέλουν να πάνε.

Έτσι η έλλειψη μιας απλής στη χρήση εφαρμογής που θα την εγκαθιστά οποιοσδήποτε εύκολα και γρήγορα στο κινητό του ανεξαρτήτως λογισμικού και θα του παρέχει όλες τις πληροφορίες σχετικά με την εύκολη και γρήγορη μετάβαση του στον προορισμό του, ήτανε η αιτία της ανάπτυξης αυτής της εφαρμογής.

## **1.2 Παρόμοιες εφαρμογές**

Υπάρχουν αρκετές εφαρμογές σχετικές με τα Αστικά Λεωφορεία για διάφορες πόλεις της Ελλάδας. Άλλες προσφέρουν μόνο βασικές λειτουργίες όπως η προβολή δρομολογίων και στάσεων ενώ άλλες περισσότερες, όπως ενημέρωση σε πραγματικό χρόνο για τις αφίξεις δρομολογίων. Χαρακτηριστικό παράδειγμα τέτοιων εφαρμογών είναι η εφαρμογή για τη πόλη της Θεσσαλονίκης και η εφαρμογή για τη πόλη της Αθήνας, Ο.Α.Σ.Θ. bus [8] και Ο.Α.Σ.Α. Telematics [9] αντίστοιχα.

Για την πόλη του Βόλου συγκεκριμένα δεν υπάρχει κάποια εφαρμογή που να σου υποδεικνύει τη βέλτιστη διαδρομή καθώς και σε πραγματικό χρόνο τις αφίξεις των δρομολογίων σε κάθε στάση.

## **1.3 Διάρθρωση της Μεταπτυχιακής Εργασίας**

Στο κεφάλαιο 2 παρουσιάζεται μια ιστορική αναδρομή για τα mobile applications και τα λειτουργικά συστήματα στα οποία δημιουργούμε εφαρμογές. Στο κεφάλαιο 3 γίνεται εκτενής αναφορά για τις υβριδικές εφαρμογές, ποια η σημασία τους, που βοηθάνε, καθώς επίσης και ποιο framework χρησιμοποιήθηκε για τη δημιουργία της εφαρμογής “VolosBus”. Στο κεφάλαιο 4 αναφέρεται ο τρόπος που σχεδιάστηκε η εφαρμογή, ποιος ήτανε ο σκοπός και οι ανάγκες που έπρεπε να καλύψει, παρουσιάζονται τα επίπεδα που δημιουργήθηκαν και πως επικοινωνούνε μέσω διαγραμμάτων. Στη συνέχεια στο κεφάλαιο 5 παρουσιάζεται η υλοποίηση της εφαρμογής, πως ικανοποιήθηκαν οι ανάγκες που προέκυψαν κατά το σχεδιασμό και πως επιτεύχθηκε το τελικό αποτέλεσμα. Παρουσιάζονται τα λογικά διαγράμματα των βασικών αλγορίθμων που συντέλεσαν στην ανάπτυξη των κυριότερων λειτουργιών της εφαρμογής για να γίνει ποιο κατανοητός ο τρόπος ανάπτυξης της εφαρμογής. Στο κεφάλαιο 6 γίνεται πλήρης επίδειξη της εφαρμογής μέσα από διάφορα screenshots κατά

την εκτέλεση της εφαρμογής σε smartphone με android λειτουργικό σύστημα. Τέλος, το κεφάλαιο 7 αξιολογεί την εφαρμογή και παρουσιάζει ιδέες για τη βελτίωση της και για μελλοντικές επεκτάσεις της και το κεφάλαιο 8 περιέχει τη βιβλιογραφία.

## 2 *Ιστορική αναδρομή των mobile applications*

Σε αυτό το κεφάλαιο γίνεται μια εκτενής αναφορά στα ποιο διαδεδομένα λειτουργικά συστήματα των κινητών τηλεφώνων. Επίσης αναλύεται η έννοια των native mobile εφαρμογών.

### 2.1 *Mobile εφαρμογές*

Τα τελευταία χρόνια τα κινητά έχουν γίνει "αναγκαία" στην καθημερινότητά μας. Εκτός από την χρήση που είχαν παλαιότερα, δηλαδή να επικοινωνούμε, έχουν γίνει πλέον σχεδόν υπολογιστές τσέπης. Αυτό είχε σαν αποτέλεσμα τη τεράστια και ραγδαία ανάπτυξη των εφαρμογών για τα smart phones. Χαρακτηριστικά το Google play store σε ελάχιστο χρόνο προσέγγισε τις 1 εκατ. εφαρμογές φτάνοντας με τη σειρά του το app store της Apple που ήταν και ο ηγέτης του χώρου που χάραξε (αρχικά τουλάχιστον) τον δρόμο σε αυτή την νέα βιομηχανία.

Πλέον τα κινητά έχουν ένα λειτουργικό σύστημα, που όσο πάει μοιάζει περισσότερο με λειτουργικό σύστημα υπολογιστή. Τα ποιο διαδεδομένα λειτουργικά συστήματα στην εποχή μας είναι:

#### 2.1.1 *To Android λειτουργικό σύστημα*

Η λέξη "Android" αναφέρεται συγκεκριμένα σε ένα λειτουργικό σύστημα για κινητά τηλέφωνα και όχι μόνο καθώς εξυπηρετεί και μία τεράστια γκάμα από tablet. Η ποικιλομορφία του όμως δεν σταματά εδώ, αφού μπορεί να τρέξει σ' όλων των ειδών τις συσκευές που υποστηρίζονται στο Linux, από κάμερες, κονσόλες παιχνιδιών και τηλεοράσεις μέχρι συστήματα αυτοκινήτων, ρολόγια και γυαλιά. Το λειτουργικό του σύστημα είναι βασισμένο στο λειτουργικό Linux το οποίο αναπτύσσεται από την Google. Είναι λογισμικό ανοιχτού κώδικα, το οποίο σημαίνει ότι οποιοσδήποτε μπορεί να πάρει τον πηγαίο κώδικα και να τον χρησιμοποιήσει όπως εκείνος θέλει.



Οι Android εφαρμογές είναι συνήθως γραμμένες σε Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού αναπτυγμένων από την Google. Η Google παρέχει το Native Development Kit (NDK), το οποίο επιτρέπει στους developer να γράψουν σημαντικό μέρος των εφαρμογών τους σε native γλώσσες, όπως είναι η C και η C++. Αυτό είναι ιδιαίτερα χρήσιμο καθώς μας επιτρέπει να επαναχρησιμοποιήσουμε τον υπάρχων κώδικα (δηλαδή κάποιες γνωστές βιβλιοθήκες) ή να βελτιώσουμε την κατανάλωση πόρων της εφαρμογής που φτιάχνουμε κάνοντας το μέγιστο δυνατό optimization (βελτιστοποίηση) στα σημεία που χρειάζεται.

Το Android είναι το πιο ευρέως διαδεδομένο λειτουργικό σύστημα στον κόσμο. Οι συσκευές με Android έχουν περισσότερες πωλήσεις από όλες τις συσκευές Windows, IOS και Mac OS X μαζί. Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιρειών, εταιρειών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις συσκευές κινητής τηλεφωνίας. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού.



*Εικόνα 1*

### **2.1.2 Το IOS λειτουργικό σύστημα**

Το IOS (προηγουμένως iPhone OS) είναι ένα λειτουργικό σύστημα για κινητά το οποίο αναπτύχθηκε και διανέμεται από την Apple Inc. Αρχικά παρουσιάστηκε το 2007 για το iPhone, ενώ υποστηρίζει και άλλες συσκευές της Apple όπως το iPod touch (Σεπτέμβριος 2007), το iPad (Ιανουάριος 2010) και το Apple TV (δεύτερης γενιάς)

(Σεπτέμβριος 2010). Αντίθετα από το Windows Phone της Microsoft και το Android της Google, η Apple δεν δίνει την άδεια για την εγκατάσταση του λογισμικού IOS σε συσκευές που δεν είναι κατασκευής Apple. Τον Ιανουάριο του 2013, το App Store της Apple περιείχε περισσότερες από 775.000 εφαρμογές IOS, 300.000 από τις οποίες ήταν συμβατές για iPad. Αυτές οι εφαρμογές έχουν μεταφορτωθεί συνολικά περισσότερα από 50 δισεκατομμύρια φορές.

Οι IOS εφαρμογές ήταν γραμμένες σε Objective C, το 2014 όμως η Apple ανακοίνωσε τη δημιουργία της Swift, μιας γλώσσας προγραμματισμού που θα λειτουργούσε ως διάδοχος της Objective C για τις εφαρμογές που κυκλοφορούσαν για την πλατφόρμα του IOS και του OS X. Με την Swift η Apple προσπάθησε να πάρει τα καλύτερα στοιχεία των πιο απλών γλωσσών προγραμματισμού, όπως είναι η JavaScript και η Python, και να δημιουργήσει μια μοντέρνα γλώσσα που θα μαθαίνεται πιο εύκολα από κάποιον που τώρα ξεκινάει τον προγραμματισμό. Ταυτόχρονα η εταιρεία υποστηρίζει πως σε γενικές γραμμές ένας αλγόριθμος θα εκτελείται πιο γρήγορα όταν είναι γραμμένος σε Swift, παρά σε Objective C. Στο παράδειγμα που έδειξε η εταιρεία κατά την παρουσίαση η διαφορά ήταν της τάξης του 30%.



*Εικόνα 2*

### ***2.1.3 Το Windows phone λειτουργικό σύστημα***

Το Windows Phone είναι η έκδοση του Λειτουργικού Συστήματος Windows της Microsoft για έξυπνα κινητά. Αναπτύχτηκε στις αρχές του 2000, ενώ στις πρώτες εκδόσεις του είχε το όνομα Windows\_Mobile. Από το 2010 μέχρι σήμερα οι πωλήσεις των Windows Phone έχουν αυξηθεί κατά πολύ ενώ υπολογίζεται από ειδικούς πως μέχρι το 2018 θα αυξηθούν κι άλλο.

Οι windows phone εφαρμογές γράφονται χρησιμοποιώντας C# / Visual Basic.NET (.NET), C++ (CX) ή HTML5/JavaScript. Οι εφαρμογές σχεδιάζονται και ελέγχονται μέσω του Visual Studio ή του Visual Studio Express.



*Εικόνα 3*

## **2.2 Native Mobile εφαρμογές**

Μια native mobile εφαρμογή όπως προ είπαμε είναι μια εφαρμογή για Smartphone η οποία έχει δημιουργηθεί σε μια συγκεκριμένη γλώσσα προγραμματισμού, όπως είναι η objective C ή η Swift για IOS λειτουργικό σύστημα ,η Java για Android περιβάλλον και η C++ για Windows phone. Δηλαδή χτίζοντας μια native εφαρμογή σημαίνει ότι δημιουργούμε μια εφαρμογή στη γλώσσα προγραμματισμού της κάθε πλατφόρμας που θέλουμε να τρέχει η εφαρμογή. Αυτό σημαίνει ότι το user interface των native εφαρμογών και η απόκρισή τους είναι πολύ γρήγορη, επίσης μπορούμε να χρησιμοποιήσουμε και όλα τα advanced χαρακτηριστικά της εκάστοτε συσκευής όπως camera, GPS, πυξίδα, γυροσκόπιο, κτλ. Οι χρήστες πολλές φορές έχουν ακόμα και τη δυνατότητα να χρησιμοποιήσουν κάποιες συσκευές χωρίς πρόσβαση στο Internet. Ένα σημαντικό όμως μειονέκτημα αυτών των εφαρμογών είναι ότι η ανάπτυξη τους κοστίζει αρκετά καθώς αν μια εταιρεία θέλει να δημιουργήσει εφαρμογές και για τα δυο βασικά λειτουργικά κινητών ή tablet (Android και IOS), θα πρέπει να δημιουργήσει τουλάχιστον 2 διαφορετικές εφαρμογές (και να δουλέψει με 2 διαφορετικά SDK). Σε αυτό το πρόβλημα έρχονται οι Hybrid Mobile εφαρμογές να δώσουν τη λύση.

## 3 *Hybrid Mobile Development*

Στο κεφάλαιο 3 αναλύεται τι είναι οι υβριδικές εφαρμογές για κινητά τηλέφωνα. Εξηγείται ο τρόπος που δημιουργούνται και πως δουλεύουν. Επίσης, παρουσιάζονται τα πιο βασικά framework που υπάρχουν και βοηθούν στη δημιουργία υβριδικών εφαρμογών. Τέλος, αναφέρεται ποιο από τα frameworks που παρουσιάστηκαν επιλέχτηκε για την ανάπτυξη της εφαρμογής και εξηγούνται οι λόγοι αυτής της επιλογής.

### 3.1 *Τι είναι το Hybrid Mobile Application*

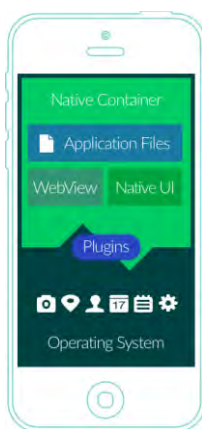
Οι υβριδικές εφαρμογές (Hybrid applications) αποτελούν ένα συνδυασμό των native και web-based εφαρμογών. Σύμφωνα, λοιπόν, με αυτή τους την ιδιότητα οι προγραμματιστές έχουν τη δυνατότητα να υλοποιήσουν τις κύριες λειτουργίες μιας εφαρμογής χρησιμοποιώντας web τεχνολογίες ενώ παράλληλα διατηρούν την πρόσβαση σε χαρακτηριστικά που ήταν διαθέσιμα μόνο σε native εφαρμογές, όπως είναι η Camera, το GPS, η πυξίδα και το γυροσκόπιο, μέσω των native APIs, που λειτουργούν ως γέφυρα ανάμεσα στις κύριες λειτουργίες της εφαρμογής και τα χαρακτηριστικά αυτά της συσκευής.

Ο κύριος λόγος που αναπτύχθηκαν αυτές οι εφαρμογές είναι ότι δίνουν τη δυνατότητα στον developer να δημιουργήσει την mobile εφαρμογή μια φορά και να μπορεί να την τρέξει σε οποιαδήποτε πλατφόρμα που υποστηρίζει μοντέρνο browser όπως android, IOS ή windows phone χωρίς κάποια έξτρα προσπάθεια. Δηλαδή ο προγραμματιστής σχεδιάζει την εφαρμογή σαν ένα κανονικό web application, ένα responsive web application το οποίο προσαρμόζεται σε διαφορετικά μεγέθη οθονών χρησιμοποιώντας components που του παρέχουν τα mobile HTML5 UI frameworks. Η πρόσβαση στα native hardware στοιχεία του κινητού όπως η Camera και το γυροσκόπιο επιτυγχάνονται μέσω native JavaScript APIs που προσφέρει το εργαλείο Cordova (Phone Gap). Δηλαδή το Apache Cordova είναι μια ανοιχτού κώδικα (open source) βιβλιοθήκη που παρέχει πρόσβαση στα χαρακτηριστικά (components) των κινητών

συσκευών, μέσω ενός συνόλου από APIs που μπορούν να κληθούν μέσω JavaScript. Η εφαρμογή μεταγλωττίζεται στα platform specific native packages, και είναι έτοιμη να εγκατασταθεί στις συσκευές ή να ανέβει στα app stores.

Οι υβριδικές εφαρμογές δημιουργούνται με παρόμοιο τρόπο όπως τα websites. Δηλαδή χρησιμοποιούν ένα συνδυασμό τεχνολογιών όπως είναι η HTML[10], CSS[11], and JavaScript[12]. Όμως, αντί να έχουν ως στόχο να δημιουργήσουν πληροφορία για αυτούς τους mobile browsers, οι hybrid applications έχουν ως στόχο ένα Web View στο Android ή UIWebView στο IOS το οποίο βρίσκεται μέσα σε ένα native container. Αυτό τους επιτρέπει να έχουν πρόσβαση στις δυνατότητες του hardware της κινητής συσκευής. Όπως προαναφέρθηκε, σήμερα οι περισσότερες υβριδικές εφαρμογές χρησιμοποιούν το Apache Cordova, μια πλατφόρμα η οποία παρέχει ένα σετ από JavaScript APIs τα οποία δίνουν τη δυνατότητα στο developer να έχει πρόσβαση στο hardware του κινητού μέσω των plug-ins, τα οποία έχουν δημιουργηθεί με native code. Στην ουσία ο developer κάνει κάποιες κλήσεις στο Cordova και αυτό είναι υπεύθυνο να καλέσει τα συστήματα του κινητού μας. Το Apache Cordova αρχικά ξεκίνησε σαν ένα project με το όνομα Phone Gap.

Αυτά τα plug-ins περιλαμβάνουν APIs για να έχουμε πρόσβαση στο γυροσκόπιο, τις επαφές της κάμερας και σε άλλες λειτουργίες της συσκευής. Υπάρχει ένας μεγάλος αριθμός plug-ins τα οποία δημιουργήθηκαν και συντηρούνται από τη developer community, τα οποία μπορούν να βρεθούν στο Apache Cordova Plugins Registry. Στην εικόνα 4 φαίνεται πως οι υβριδικές εφαρμογές καταφέρνουν να έχουν πρόσβαση στα hardware χαρακτηριστικά της συσκευής.



Εικόνα 4

Αυτή η ικανότητα των υβριδικών εφαρμογών, να συνδυάζουν standard web apps με native code μπορεί να μειώσει σημαντικά το χρόνο και το κόστος ανάπτυξης της εφαρμογής καθώς και να μειώσει το coding work. Για το χρήστη, μια καλά σχεδιασμένη hybrid εφαρμογή δεν έχει καμία διαφορά από μια native εφαρμογή. Ο χρήστης κατεβάζει την εφαρμογή από το AppStore την εγκαθιστά και την τρέχει όπως θα έκανε με κάθε άλλη native εφαρμογή. Αλλά για το developer υπάρχει μια τεράστια διαφορά, αντί να γράψει ολόκληρο το κώδικα σε κάποια άλλη γλώσσα έτσι ώστε να είναι συμβατή με άλλη πλατφόρμα αρκεί να γράψει ένα μεγάλο κομμάτι κώδικα χρησιμοποιώντας html CSS, and JavaScript και να το επαναχρησιμοποιήσει σε οποιαδήποτε συσκευή θέλει, είτε αυτή είναι android είτε IOS ή windows phone.

## 3.2 *Hybrid Mobile Frameworks*

Καθώς η τεχνολογία αναπτύσσεται με γρήγορους ρυθμούς, έχει ήδη αναπτυχτεί ένας μεγάλος αριθμός εργαλείων frameworks έτσι ώστε να κάνουν τη δουλειά των developer πιο εύκολη και να παρέχουν μια μοναδική εμπειρία στους χρήστες. Η αλήθεια είναι πως το hybrid mobile app framework, δίνοντας τη δυνατότητα δημιουργίας εφαρμογών ανεξαρτήτως πλατφόρμας, έχει καταλάβει μια σημαντική θέση στην αγορά και αυτό φαίνεται με τη δημιουργία συνεχώς όλων και περισσότερων framework. Έτσι τα τελευταία χρόνια δημιουργούνται όλο και περισσότερα Hybrid Mobile Frameworks, παρέχοντας τη δυνατότητα στους developer να αναπτύσσουν καλύτερες, πιο αποτελεσματικές και αποδοτικές υβριδικές εφαρμογές. Μερικά από τα πιο διαδεδομένα frameworks για υβριδικές εφαρμογές σήμερα είναι:

### 3.2.1 *Ionic*

Το ionic είναι ένα open source mobile SDK για τη δημιουργία hybrid native mobile applications, χρησιμοποιώντας HTML, JavaScript, και CSS. Το Ionic δημιουργήθηκε από την Drifty Co. το 2013 και αυτό που το κάνει μοναδικό είναι η AngularJS και το γεγονός ότι υποστηρίζει SASS (Syntactically Awesome Style Sheets) , μια επέκταση της CSS η οποία δίνει τη δυνατότητα στο developer να δημιουργήσει πιο πλούσιες

και interactive εφαρμογές. Η AngularJS είναι ένα framework MVC(model view control) το οποίο επιτρέπει στο developer να κάνει extend τη σύνταξη της HTML και να χρησιμοποιήσει κάποια application components ποιο εύκολα. Δηλαδή, το AngularJS δένει την HTML (views) σε αντικείμενα JavaScript (models). Δηλαδή όταν συμβαίνει μια αλλαγή στα models, η σελίδα ανανεώνεται αυτόματα. Συμβαίνει επίσης και το αντίθετο, όταν δηλαδή ένα text field ενημερώνεται στο view, το AngularJS χειρίζεται την κατάσταση αυτόματα χωρίς να χρειάζεται να γίνει κάποια επιπλέον ενέργεια στην HTML με την σύνδεση κάποιου event.

### 3.2.2 *Sencha Touch*

Το Sencha Touch [13] πρόκειται για ένα υψηλής απόδοσης framework ανοιχτού κώδικα για ανάπτυξη HTML5 web εφαρμογών για Android, iPhone, Windows Phone και άλλες συσκευές. Το Sencha Touch δημιουργήθηκε από τη Sencha, η πρώτη έκδοση ήταν η 0.90 beta, και ήταν διαθέσιμη 17 Ιουλίου 2010. Το Sencha Touch παρέχει διάφορες δυνατότητες στην κατασκευή του user interface με διάφορα plug-ins, themes και άλλα χαρακτηριστικά. Πλέον υπάρχει πλήρης υποστήριξη για τη δημιουργία hybrid native εφαρμογών με τη χρήση του Apache Cordova.

### 3.2.3 *Onsen UI*

Το Onsen UI [14] δημιουργήθηκε από την Monaca, Inc. / Asial Corporation το 2013 και είναι παρόμοιο με το Ionic αν και έχει πολύ μικρότερο community support. Είναι open source και εύκολο στη χρήση και από τα πιο γρήγορα hybrid mobile frameworks. Το Onsen UI μπορεί να δουλέψει με ή χωρίς AngularJS, και έχει ένα μεγάλο documentation το οποίο περιλαμβάνει πάρα πολλά παραδείγματα και layouts για τις πιο κοινές δομές εφαρμογών. Ένα μειονέκτημα του Onsen UI είναι ότι προσφέρει μόνο IOS θέματα, αν και η επόμενη έκδοση αναμένεται να έχει περισσότερη υποστήριξη σε σχεδιαστικό υλικό.

### 3.2.4 *Kendo UI*

Το Kendo UI [15] δημιουργήθηκε από την Telerik το 2012 και είναι ένα άλλο framework για τη δημιουργία οποιασδήποτε web ή hybrid mobile εφαρμογής με

HTML, JavaScript και CSS. Βασίζεται πάνω στο jQuery και “έρχεται” πακεταρισμένο με πάνω από 70 έτοιμα για χρήση jQuery widgets. Είναι open source αν και ελαφρά περιορισμένο όσον αφορά τα χαρακτηριστικά όπως επίσης δεν υπάρχει τεχνική υποστήριξη. Το kendo UI είναι αρκετά δημοφιλής ανάμεσα στις εταιρείες. Έχει μια μεγάλη βάση πελατών συμπεριλαμβάνοντας εταιρείες όπως είναι η Sony, η Nasa, η Microsoft, η Volvo και η Toshiba.

### 3.2.5 *jQuery Mobile*

Το jQuery Mobile [16] δημιουργήθηκε από τη JS foundation 16 Οκτωβρίου 2010 και πρόκειται για ένα client-side framework για κινητές συσκευές και οθόνες αφής. Πρόκειται για μια βιβλιοθήκη περιβάλλοντος χρήστη που παρέχει τα κατάλληλα εργαλεία για την εύκολη και γρήγορη ανάπτυξη του περιβάλλοντος της εφαρμογής. Στηρίζετε στην HTML5 και στο jQuery JavaScript framework. Ο στόχος αυτού του project είναι να παρέχει υποστήριξη σε όσο το δυνατόν περισσότερες συσκευές. Το jQuery Mobile είναι ανοιχτού κώδικα και υποστηρίζεται από πολλές και μεγάλες εταιρείες. Στο user interface μπορούν να χρησιμοποιηθούν πολλά extensions και widgets από τρίτους. Είναι αρκετά διαδεδομένα μιας και στηρίζεται στο πολύ γνωστό jQuery και έχει και πολύ καλό documentation και community support. Ωστόσο είναι αρκετά αργό και παλιό καθώς η τελευταία του έκδοση βγήκε το 2014.

### 3.2.6 *Adobe PhoneGap*

Το Adobe PhoneGap [17] πρόκειται για ένα εργαλείο που επιτρέπει τη μετατροπή μιας web εφαρμογής σε native. Στηρίζεται στο Apache Cordova το οποίο αρχικά αναπτύχθηκε από την Adobe και πρόκειται για ένα project ανοιχτού κώδικα. Εκτός από το “πακετάρισμα” της web εφαρμογής σε native, πράγμα το οποίο προσφέρει δυνατότητα ανεβάσματος στο store εφαρμογών της αντίστοιχης πλατφόρμας, το PhoneGap προσφέρει και διεπαφές σε JavaScript πλήθος native δυνατοτήτων που μπορεί να μην καλύπτονται από την HTML5. Για παράδειγμα μπορούμε να έχουμε πρόσβαση στον τοπικό αποθηκευτικό χώρο της συσκευής όπως στην SD κάρτα και στο



native σύστημα ειδοποιήσεων. Οι πλατφόρμες τις οποίες υποστηρίζει είναι Android, iOS, Windows Phone, BlackBerry, Windows 8 και Tizen. Επίσης υπάρχουν διαθέσιμα πολλά plug-ins που προσφέρουν επιπρόσθετες υπηρεσίες ενώ η Adobe έχει ειδική υπηρεσία που μετατρέπει στο Cloud το HTML5 web application σε native σε όλες τις υποστηριζόμενες πλατφόρμες με το Adobe PhoneGap Build.

### 3.3 Λόγοι επιλογής του *Ionic*

Το Ionic τα τελευταία χρόνια έχει εδραιωθεί ως ο ηγέτης των hybrid mobile frameworks. Η ομάδα του ionic διατηρεί το framework αναβαθμισμένο προσαρμόζοντάς το συνεχώς στα τελευταία trends αυξάνοντας έτσι τον ανταγωνισμό. Ένα από τα μεγαλύτερα πλεονεκτήματα του ionic είναι ότι είναι ανοιχτού κώδικα, η κοινότητα του ionic προσφέρει ένα τεράστιο φάσμα πηγών δηλαδή επιτρέπει να διαμοιράζονται ελεύθερα διαφορετικά πρότυπα (templates) και τεχνικές για τη δημιουργία εφαρμογών κάνοντας την ανάπτυξη πολύπλοκων εφαρμογών πολύ πιο εύκολη. Το Ionic «έρχεται» με JavaScript πρότυπα για να βοηθήσει να οικοδομήσουμε κάποιες σοβαρές εφαρμογές με παρόμοια χαρακτηριστικά με τις εφαρμογές του IOS και Android. Ως σκοπό έχει να επιτρέπει στους προγραμματιστές να δημιουργούν το ίδιο είδος των ισχυρών αλληλεπιδράσεων User Interface, που συναντάμε στις native εφαρμογές, όπως είναι τα γραφικά και τα πλευρικά μενού πλοήγησης. Επιπλέον ένας ακόμη σκοπός ήταν να ελευθερωθούν οι υβριδικές εφαρμογές από το URL-bar και να κινηθούν προς πιο γενικές και πιο ισχυρές View Controller έννοιες. Το Ionic είναι το κομμάτι που λείπει κατά τη δημιουργία native εφαρμογών με web standards. Με τη χρήση CSS και JavaScript, επιτυγχάνεται η γρήγορη επιτάχυνση της ανάπτυξης native εφαρμογών με HTML5. Έτσι όταν ο χρήστης είναι έτοιμος να προωθήσει την εφαρμογή προς AppStores, πραγματοποιείται compile της εφαρμογής με το Cordova και έχουμε μια εφαρμογή που θα τρέξει για τις πιο δημοφιλείς πλατφόρμες. Συνοπτικά μερικά από τα πιο βασικά πλεονεκτήματα του ionic είναι:

- ο Μεγάλη developer community
- ο Command Line Interface με πολλά χρήσιμα χαρακτηριστικά
- ο Ανοιχτού κώδικα και ελεύθερο (κάτω από μια ειδική MIT άδεια)
- ο Χτίζεται γύρω από την **AngularJS/AngularJS2**

- Υποστηρίζει iOS, Android, Mobile Windows
- Επικεντρώνεται στη δημιουργία native/hybrid mobile apps παρά σε mobile websites
- Υποστηρίζει **Cordova**, **PhoneGap**, ή **Trigger.io**



*Εικόνα 5*

## 4 Σχεδιασμός Εφαρμογής

Σε αυτό το κεφάλαιο γίνεται μια αναλυτική εξαγωγή απαιτήσεων και καθορισμός προδιαγραφών που αφορούν την εφαρμογή “VolosBus”. Περιγράφονται αναλυτικά οι ανάγκες που οδήγησαν στην υλοποίηση της εφαρμογής και τέλος περιγράφεται ο σχεδιασμός της και το διάγραμμα ροής.

### 4.1 Σκοπός και ανάγκες κάλυψης της εφαρμογής

Όπως προαναφέρθηκε και σε παραπάνω κεφάλαιο υπήρχε έλλειψη μιας εφαρμογής που θα βοηθούσε τους κατοίκους και επισκέπτες του Βόλου να μετακινούνται ποιο εύκολα μέσα στη πόλη. Έτσι η απαίτηση ήταν να δημιουργηθεί μια εφαρμογή που ο βασικός σκοπός της θα ήταν η διευκόλυνση των χρηστών κατά τη μετακίνηση τους μέσα στη πόλη με την Αστική Συγκοινωνία. Μια εφαρμογή που θα πρόσφερε πληροφορίες άμεσα με βάση τον πραγματικό χρόνο και την ακριβή τοποθεσία του χρήστη.

Μια μεγάλη ανάγκη που είχε σκοπό να καλύψει η εφαρμογή ήταν η δημιουργία μιας εφαρμογής που θα ήταν εύκολη στη χρήση και θα μπορούσε να τη χρησιμοποιήσει οποιοσδήποτε ανεξαρτήτως ηλικίας και χωρίς καμία περεταίρω γνώση και εξειδίκευση. Θα έπρεπε δηλαδή να δημιουργηθεί μια εφαρμογή που θα μπορούσε να τη χρησιμοποιήσει με την ίδια ευκολία τόσο κάποιος που κατέχει γνώσεις πάνω στις τεχνολογίες και τις εφαρμογές μέχρι και κάποιος ίσως μεγαλύτερης ηλικίας που δεν θα είχε καμία περαιτέρω γνώση πάνω στον προγραμματισμό.

Πέρα από τον χρήστη όμως υπήρχε και η ανάγκη να δημιουργηθεί μια εφαρμογή με μειωμένο κόστος ανάπτυξης και συντήρησης. Για παράδειγμα, θα έπρεπε να δημιουργηθεί μια εφαρμογή γραμμένη σε μια γλώσσα που θα μπορούσε να είναι συμβατή τόσο σε Android όσο και σε IOS και Windows phone κινητά χωρίς να χρειάζεται μεγάλο κόστος για την παραγωγή σε κάθε ένα ξεχωριστά. Γιατί μια εφαρμογή που θα ήταν αποκλειστικά γραμμένη για android θα έπρεπε να ξαναγραφτεί σε άλλη γλώσσα ώστε να είναι συμβατή με IOS κινητά και σε άλλη για να μπορεί να

εγκατασταθεί σε windows phones με αποτέλεσμα να αυξάνεται υπερβολικά το κόστος παραγωγής της. Επίσης δεν έπρεπε να αμελήσουμε και το κόστος συντήρησης. Όπως αυξάνεται το κόστος παραγωγής έτσι αυξάνεται και το κόστος συντήρησης γιατί θα έπρεπε να συντηρούνται παραπάνω από μια εκδόσεις της εφαρμογής.

Όσον αφορά τις ανάγκες που έπρεπε να καλύψει η εφαρμογή θα έπρεπε να πληροφορεί τον χρήστη για τα δρομολόγια και τις στάσεις που περνάει το κάθε λεωφορείο. Όπως επίσης και για τις ώρες αφίξεων λεωφορείων από την αφετηρία τους. Άλλη μια ανάγκη που έπρεπε να καλυφθεί ήταν να βρεθεί τρόπος να ενημερώνεται ο χρήστης ανάλογα με το προορισμό του ποια γραμμή να επιλέξει, σε ποια στάση θα έπρεπε να μεταβεί και σε ποια να αποβιβαστεί. Επειδή όμως η πόλη του Βόλου είναι αρκετά μεγάλη και υπάρχουν αρκετά λεωφορεία που διανύουν όλη τη πόλη αυτή η ανάγκη δημιούργησε μια άλλη. Θα έπρεπε η εφαρμογή να σχεδιαστεί έτσι ώστε να λαμβάνει υπόψη τις ανταποκρίσεις των λεωφορείων και άμα ένας προορισμός ήταν αρκετά μακριά και δεν καλύπτονταν μόνο με ένα λεωφορείο θα έπρεπε να προτείνεται στο χρήστη περισσότερες από μια γραμμές. Δηλαδή θα έπρεπε να σχεδιαστεί με ένα τρόπο ώστε ανάλογα με το προορισμό που θα δήλωνε ο χρήστης να τον ενημέρωνε σε ποια στάση θα έπρεπε να μεταβεί, σε ποια γραμμή θα έπρεπε να μεταβιβαστεί πρώτα και σε ποια στη συνέχεια επίσης θα έπρεπε να τον ενημερώσει σε ποιες στάσεις θα έπρεπε να αποβιβαστεί. Όπως επίσης και για τις αποστάσεις που έπρεπε να διανύσει με τα πόδια. Καθώς μια ανάγκη “γεννάει” μια άλλη και η ανάγκη να αναπτυχθεί ένας τρόπος να πληροφορεί τον χρήστη ποιες γραμμές να επιλέξει για να φτάσει στον προορισμό του γέννησε την ανάγκη να δημιουργηθεί ένας τρόπος ώστε αν χρειάζεται ο χρήστης να πάρει δύο γραμμές για να μεταβεί στον προορισμό του και είναι εφικτό η μια να διανυθεί με τα πόδια, να του δίνεται η επιλογή μιας οικονομικότερης διαδρομής, όπου θα μπορεί να επιλέξει να μεταβεί στον προορισμό του επιβαίνοντας σε ένα λεωφορείο ή να επιλέξει μια πιο ακριβή διαδρομή αλλά χωρίς να χρειαστεί να περπατήσει πολλά μέτρα, επιβαίνοντας σε δυο λεωφορεία. Για να είναι όλα αυτά ακόμα πιο κατανοητά θα έπρεπε να δίνεται η δυνατότητα στο χρήστη να ενημερώνεται για όλη αυτή τη πληροφορία τόσο βλέποντας την πορεία και τις στάσεις πάνω στο χάρτη όσο και να μπορεί να τη διαβάσει με λόγια. Θα έπρεπε δηλαδή ο χρήστης να έβλεπε πάνω στο χάρτη που βρίσκεται, σε ποιες στάσεις θα πρέπει να μεταβεί και να αποβιβαστεί και

την πορεία που θα κάλυπτε με τα λεωφορεία. Όπως επίσης έπρεπε να του δίνεται η δυνατότητα να μπορεί να διαβάσει με λόγια όλες τις οδηγίες.

Μια ακόμα απαίτηση που έπρεπε να καλύψει η εφαρμογή ήταν η δυνατότητα αναζήτησης των πλησιέστερων στάσεων και εκδοτηρίων με βάση τη θέση του χρήστη, η ενημέρωση των αποστάσεων από κάθε στάση και εκδοτήριο εισιτηρίων καθώς και η εμφάνιση τους πάνω στο χάρτη. Επίσης θα έπρεπε να πληροφορεί το χρήστη για τα ωράρια λειτουργίας των εκδοτηρίων έτσι ώστε άμα το ποιο κοντινό τη συγκεκριμένη ώρα ήταν κλειστό να μετέβαινε στο επόμενο.

Άλλο ένα σημαντικό πρόβλημα είναι η αναμονή σε μια στάση για ένα λεωφορείο χωρίς να ξέρει ο πολίτης σε πόση ώρα αναμένεται το λεωφορείο που τον ενδιαφέρει. Είναι γεγονός ότι είναι πολύ σημαντικό να ξέρεις σε πόση ώρα αναμένεται το λεωφορείο που περιμένεις ώστε να κανονίζεις να μεταβείς την ώρα που πρέπει χωρίς να χάνεις ώρα απλά περιμένοντας σε μια στάση. Έτσι η εφαρμογή έπρεπε να σχεδιαστεί με ένα τρόπο ώστε να δίνεται η δυνατότητα στο χρήστη να ενημερώνεται ανάλογα με τη στάση που τον ενδιαφέρει σε πόση ώρα θα περάσει ένα συγκεκριμένο λεωφορείο και σε πόση ώρα αναμένεται το επόμενο.

Τέλος, θα πρέπει να δίνεται η δυνατότητα στο χρήστη να ενημερώνεται για όλα τα βασικά θέματα και να μαθαίνει τα νέα που αφορούν την αστική συγκοινωνία μέσα από την εφαρμογή.

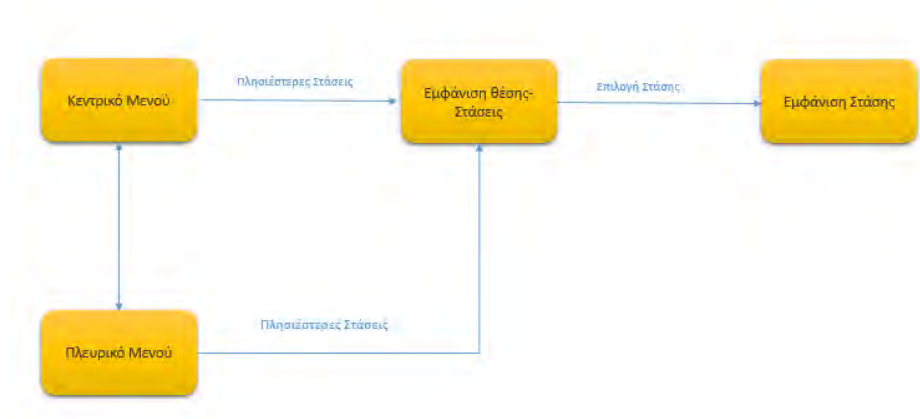
## **4.2 Σχεδιασμός λειτουργιών εφαρμογής**

Οι βασικές λειτουργίες που θα πρέπει να δίνονται στο χρήστη θα είναι οι εξής:

### **1. Πλησιέστερες στάσεις**

Θα εντοπίζεται με GPS η θέση του χρήστη και θα εμφανίζεται πάνω στο χάρτη, επίσης με βάση τις συντεταγμένες του χρήστη και των στάσεων θα υπολογίζονται οι αποστάσεις και θα ταξινομούνται οι στάσεις με βάση τη ποιο κοντινή στο χρήστη. Οι στάσεις θα εμφανίζονται ταξινομημένες κάτω από το χάρτη που θα δείχνει τη θέση του χρήστη με περαιτέρω πληροφορίες όπως το όνομα της στάσης και σε πόσα μέτρα βρίσκεται από τη θέση του χρήστη καθώς

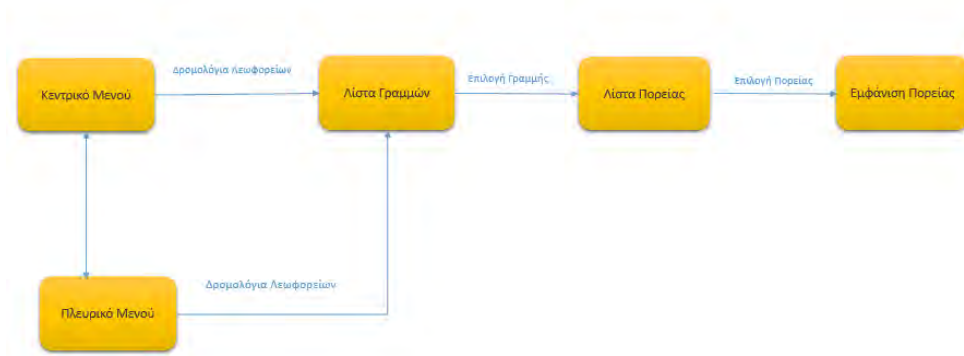
και προς τα πού πηγαίνει το λεωφορείο που ανέρχεται από αυτή. Ο χρήστης θα πρέπει να έχει τη δυνατότητα όταν πατάει πάνω στη στάση που τον ενδιαφέρει να τη βλέπει πάνω στο χάρτη. Η ροή αυτής τη λειτουργίας φαίνεται στην εικόνα 6.1



Εικόνα 6.1

## 2. Δρομολόγια λεωφορείων

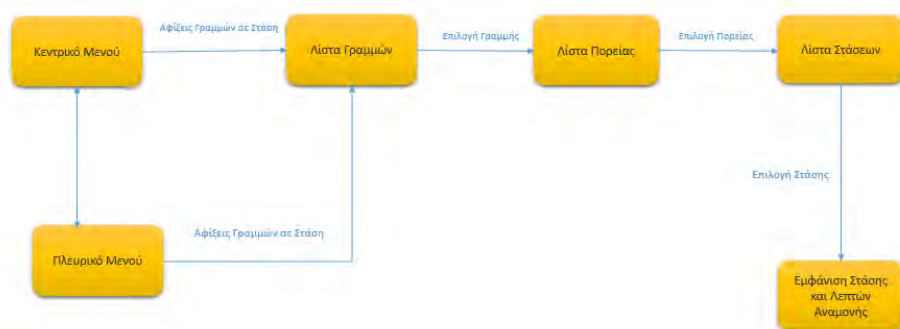
Άλλη μια λειτουργία που θα πρέπει να προσφέρει η εφαρμογή είναι να δίνεται η δυνατότητα στο χρήστη να βλέπει τη πορεία και όλες τις στάσεις που περνάει κάθε γραμμή που τον ενδιαφέρει. Δηλαδή θα δίνεται μια δυνατότητα στο χρήστη να επιλέγει όποια γραμμή θέλει από μια λίστα με όλα τα αστικά λεωφορεία του Δήμου Βόλου και αφού κάνει την επιλογή του, θα πρέπει να εμφανίζεται πάνω στο χάρτη η πορεία και οι στάσεις καθώς επίσης και τα ονόματα των στάσεων της κάθε γραμμής που επιλέγει. Η ροή της λειτουργίας που περιγράφεται παραπάνω θα πρέπει να είναι όπως απεικονίζεται στην εικόνα 6.2



Εικόνα 6.2

### 3. Αφίξεις

Άλλη μια απαίτηση που πρέπει να καλύψει η εφαρμογή και η ροή της οποίας φαίνεται στην εικόνα 6.3 είναι η δυνατότητα να ενημερώνει το χρήστη για την ώρα των αφίξεων οποιουδήποτε λεωφορείου σε κάθε στάση. Για παράδειγμα, ο χρήστης θα έχει τη δυνατότητα να επιλέγει όποια γραμμή τον ενδιαφέρει από μια λίστα με τα προσφερόμενα αστικά λεωφορεία του Βόλου και να εμφανίζονται στην οθόνη του όλες οι στάσεις που περνάει το εκάστοτε λεωφορείο. Στη συνέχεια, ο χρήστης θα πρέπει να μπορεί επιλέγοντας μια στάση να δει σε πόσα λεπτά αναμένεται να περάσουν τα δύο επόμενα λεωφορεία από τη συγκεκριμένη στάση καθώς επίσης και ποιες άλλες γραμμές περνάνε από το συγκεκριμένη στάση και σε πόση ώρα. Επίσης θα πρέπει ο χρήστης να μπορεί να δει στην οθόνη του την κάθε στάση μέσω street view.



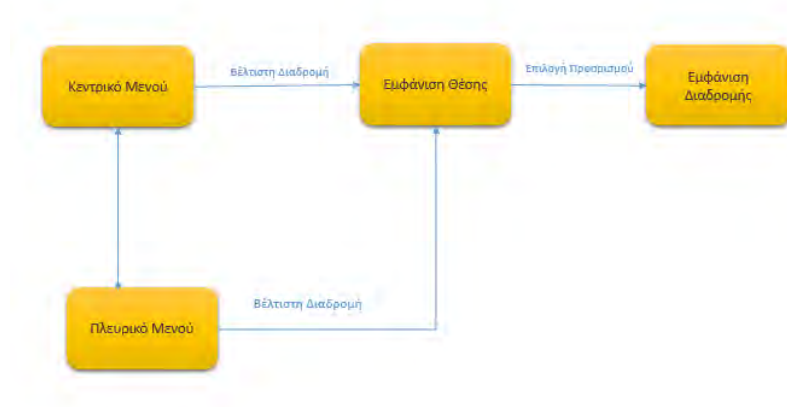
Εικόνα 6.3

#### 4. Βέλτιστη διαδρομή

Θα εντοπίζεται με GPS η θέση του χρήστη και θα εμφανίζεται πάνω στο χάρτη. Κάπου στην οθόνη θα δίνεται η δυνατότητα στο χρήστη να εισάγει τη διεύθυνση προορισμού του ή κάποιο άλλο σημείο ενδιαφέροντος που υπάρχει πάνω στους χάρτες της google. Στη συνέχεια θα πρέπει να του εμφανίζεται στην οθόνη η πορεία που θα χρειαστεί να διανύσει τόσο περπατώντας όσο και επιβαίνοντας στο λεωφορείο για να φτάσει στον προορισμό του. Επίσης θα πρέπει να δίνεται η δυνατότητα να μπορεί ο κάθε χρήστης να διαβάσει τη πληροφορία αυτή και με λόγια. Για παράδειγμα, η εφαρμογή θα πρέπει να πληροφορεί το χρήστη από το σημείο που βρίσκεται σε ποια στάση να μεταβεί και σε πόσα μέτρα, ποια γραμμή να επιλέξει και που να αποβιβαστεί. Σε περίπτωση που μόνο μια γραμμή δεν καλύπτει τον προορισμό του χρήστη θα πρέπει η εφαρμογή να προτείνει και δεύτερη γραμμή, δηλαδή, σε ποια στάση να αποβιβαστεί ο χρήστης αφού έχει πάρει το πρώτο λεωφορείο και σε ποια να μεταβεί ώστε να πάρει το επόμενο. Μια βασική απαίτηση από την εφαρμογή είναι να κρίνει και να εμφανίζει στον χρήστη μια ή περισσότερες επιλογές δρομολογίου, δηλαδή να κρίνει πότε να του εμφανίσει μόνο μια επιλογή διαδρομής (παίρνοντας ένα ή περισσότερα λεωφορεία) και πότε να του εμφανίσει δυο επιλογές, μια οικονομικότερη παίρνοντας ένα λεωφορείο και να διανύσει την υπόλοιπη απόσταση περπατώντας, και μια δεύτερη επιλογή παίρνοντας δύο λεωφορεία για να φτάσει στον προορισμό του. Επίσης θα



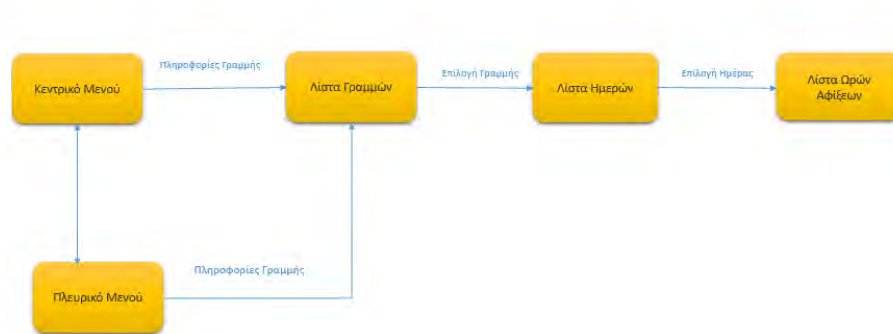
πρέπει ο χρήστης να πληροφορείται για τα μέτρα κάθε απόστασης που θα πρέπει να διανύσει είτε περπατώντας είτε με το λεωφορείο. Στο παρακάτω σχήμα φαίνεται η διαγραμματική ροή αυτής της λειτουργίας.



Εικόνα 6.4

## 5. Γραμμές

Θα εμφανίζεται στο χρήστη μια λίστα με όλες τις γραμμές, στη συνέχεια αφού επιλέξει ο χρήστης ποια γραμμή τον ενδιαφέρει θα πρέπει να του δίνεται η δυνατότητα να επιλέξει και τη μέρα που θέλει να δει τις ώρες αφίξεων των γραμμών από την αφετηρία τους. Αφού γίνει η επιλογή της γραμμής και της μέρας, θα πρέπει να εμφανίζονται στην οθόνη δυο λίστες με τις ώρες αφίξεων της γραμμής, μια λίστα με τις ώρες μετάβασης και μια για τις ώρες επιστροφής. Επίσης θα πρέπει να υπάρχει η δυνατότητα να ενημερώνεται ο χρήστης για την άμεση άφιξη του λεωφορείου που τον ενδιαφέρει με βάση την ώρα εκείνης της στιγμής, επισημαίνοντας τη συγκεκριμένη ώρα με κάποιο διαφορετικό χρώμα ή ποιο έντονα τονισμένα γράμματα, έτσι ώστε να γίνεται κατανοητό στο χρήστη ποια είναι η ακριβής ώρα που αναμένεται να αναχωρήσει το επόμενο λεωφορείο με βάση την ώρα εκείνης της στιγμής που κοιτάει την εφαρμογή. Η ροή αυτής της λειτουργίας απεικονίζεται στο σχήμα 6.5.



Εικόνα 6.5

#### 6. **Κοντινότερο εκδοτήριο εισιτηρίων**

Θα εντοπίζεται με GPS η θέση του χρήστη και θα εμφανίζεται πάνω στο χάρτη, επίσης με βάση τις συντεταγμένες του χρήστη και των εκδοτηρίων εισιτηρίων θα υπολογίζονται οι αποστάσεις και θα ταξινομούνται τα εκδοτήρια με βάση το ποιο κοντινό στο χρήστη. Θα πρέπει να εμφανίζονται όλα τα εκδοτήρια ταξινομημένα κάτω από το χάρτη που θα δείχνει τη θέση του χρήστη με επιπλέον πληροφορίες όπως η διεύθυνση που βρίσκεται το εκδοτήριο, αν είναι αυτόματος πωλητής εισιτηρίων ή εκδοτήριο, στα πόσα μέτρα βρίσκεται από τη θέση του χρήστη καθώς επίσης και τις ώρες λειτουργίας του. Επίσης ο χρήστης θα πρέπει να έχει τη δυνατότητα όταν πατάει πάνω στο εκδοτήριο που τον ενδιαφέρει να βλέπει που βρίσκεται πάνω στο χάρτη. Παρακάτω απεικονίζεται η διαγραμματική ροή της λειτουργίας



Εικόνα 6.6

## 7. Νέα

Η εφαρμογή θα πρέπει να δίνει τη δυνατότητα στο χρήστη να πληροφορείται για όλα τα τελευταία νέα που αφορούν την αστική συγκοινωνία του Βόλου. Η βασική ροή αυτής της λειτουργίας θα έχει τη μορφή που δείχνει το διάγραμμα στην εικόνα 6.7



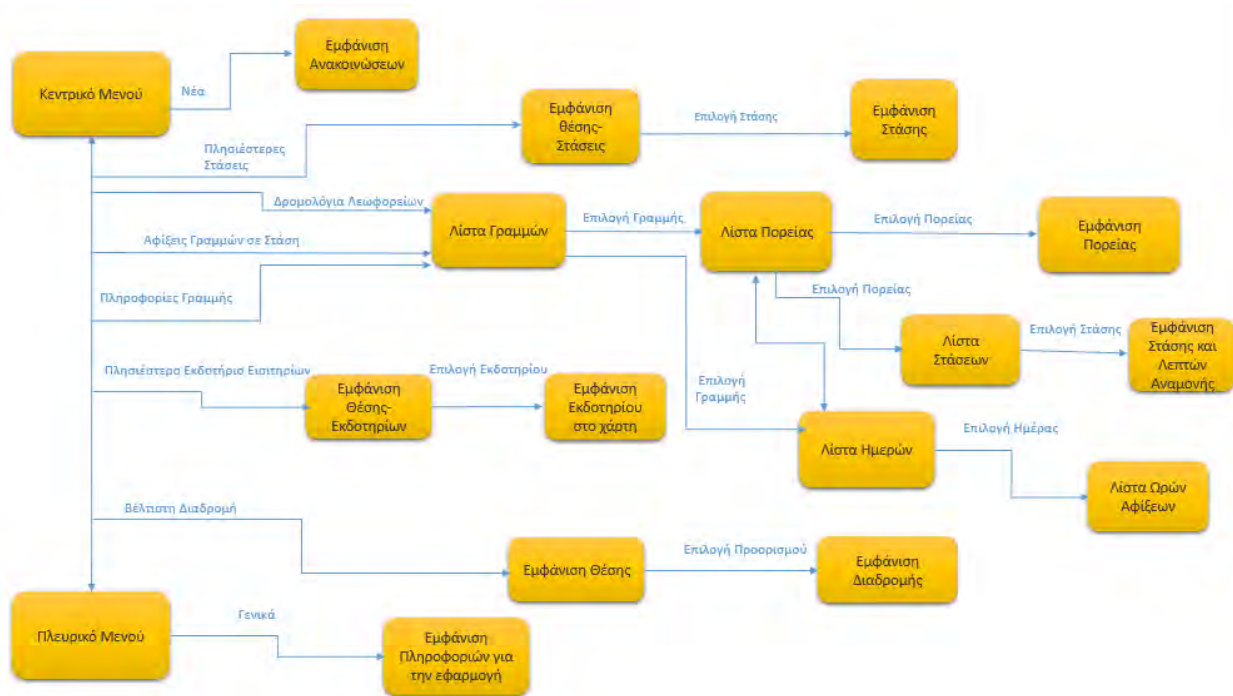
Εικόνα 6.7

Όπως αναφέρθηκε και παραπάνω, όλες αυτές οι λειτουργίες που έχουν σχεδιαστεί να προσφέρονται μέσω της εφαρμογής θα πρέπει να δίνονται στο χρήστη με εύκολο και γρήγορο τρόπο. Ο χρήστης θα πρέπει να μπορεί να πλοηγεί μέσα στην εφαρμογή γρήγορα και εύκολα μεταβαίνοντας από τη μια λειτουργικότητα στην άλλη χωρίς να χρειάζεται να γυρίζει στην πρώτη σελίδα της εφαρμογής, και αυτό θα επιτυγχάνεται είτε μέσω κάποιου tab μέσα από τη

σελίδα που βρίσκεται ήδη είτε μέσω του πλευρικού μενού που θα πρέπει να υπάρχει στην εφαρμογή για περαιτέρω διευκόλυνση του χρήστη.

### 4.3 Διάγραμμα ροής

Κατά την ανάλυση και το σχεδιασμό των επιμέρους λειτουργιών που θα καλύπτει η εφαρμογή δημιουργήθηκε ένα διάγραμμα που απεικονίζει ολόκληρη τη ροή λειτουργίας της εφαρμογής όπως φαίνεται στην εικόνα 7 έτσι ώστε να γίνει πιο κατανοητή η ροή και πιο εύκολη η υλοποίηση της εφαρμογής. Όπως φαίνεται στο παρακάτω διάγραμμα η εφαρμογή θα αποτελείται από κάποια βασικά επίπεδα που θα καλύπτουν τις βασικές λειτουργίες όπως είναι εμφάνιση της λίστας γραμμών και στάσεων και από εκεί και πέρα ανάλογα τι θα έχει επιλέξει ο χρήστης θα μεταβαίνει στην ανάλογη λειτουργία.



Εικόνα 7

Πέρα από τις βασικές λειτουργίες όπως είναι η απλή ενημέρωση των χρηστών για νέα, η ενημέρωση για κοντινές στάσεις και εκδοτήρια εισιτηρίων, υπάρχουν και πιο πολύπλοκες στη ροή λειτουργίες, όπως είναι οι αφίξεις γραμμών σε στάση, τα δρομολόγια λεωφορείων και οι πληροφορίες γραμμής, όπου ο χρήστης όπως φαίνεται

και από το διάγραμμα αφού έχει επιλέξει τη γραμμή που τον ενδιαφέρει να μπορεί να δει και άλλες λειτουργίες πέρα από αυτή που επέλεξε αρχικά χωρίς να χρειάζεται να μεταβεί πάλι στο κεντρικό μενού και να κάνει άλλη επιλογή. Για παράδειγμα, αν βρίσκεται στο κεντρικό μενού και θέλει να δει τα δρομολόγια θα πρέπει να του εμφανίζεται μια λίστα από όλα τα αστικά λεωφορεία του Βόλου. Από εκεί θα μπορεί να επιλέξει τη γραμμή που τον ενδιαφέρει και σε εκείνο το σημείο θα πρέπει να μπορεί να δει για τη συγκεκριμένη γραμμή, πέρα από τη πορεία του δρομολογίου που έχει επιλέξει αλλά και τις λίστες ωρών αφίξεων και τις στάσεις που διαπερνάει η συγκεκριμένη γραμμή.

## 5 Υλοποίηση της εφαρμογής

Σε αυτό το κεφάλαιο γίνεται παρουσίαση της υλοποίησης της εφαρμογής, δηλαδή εξηγείται πως ικανοποιήθηκαν οι ανάγκες που προέκυψαν και παρουσιάστηκαν στο προηγούμενο κεφάλαιο κατά το σχεδιασμό της υλοποίησης, καθώς επίσης και πως επιτεύχθηκε το τελικό αποτέλεσμα. Τέλος, παρουσιάζονται οι βασικοί αλγόριθμοι της εφαρμογής και επεξηγούνται με λογικά διαγράμματα.

### 5.1 Λεπτομέρειες υλοποίησης

Στο προηγούμενο κεφάλαιο κατά το σχεδιασμό παρουσιάστηκαν ο σκοπός και οι ανάγκες υλοποίησης της εφαρμογής. Όπως προαναφέρθηκε μια από της βασικότερες ανάγκες ήταν η υλοποίηση μιας εφαρμογής εύκολη στη χρήση από τον οποιονδήποτε ανεξαρτήτως λειτουργικού συστήματος που κατέχει, δηλαδή, μια εφαρμογή συμβατή με όλα τα λειτουργικά συστήματα κινητών χωρίς όμως να αυξάνεται παράλληλα το κόστος ανάπτυξης και κατά συνέπεια το κόστος συντήρησης της εφαρμογής. Όλες αυτές οι ανάγκες καλύφθηκαν δημιουργώντας όπως έχει προαναφερθεί μια υβριδική εφαρμογή. Έπειτα από μελέτη των διάφορων υβριδικών framework που κυκλοφορούνε αποφασίστηκε η εφαρμογή να δημιουργηθεί με τη βοήθεια του ionic. Οι λόγοι αυτής της απόφασης περιγράφονται αναλυτικά σε προηγούμενο κεφάλαιο βλ. Παράγραφος 3. Η εφαρμογή λοιπόν γράφθηκε σε HTML, χρησιμοποιώντας JavaScript και CSS. Με τη βοήθεια του εργαλείου Cordova δόθηκε πρόσβαση στα hardware χαρακτηριστικά της συσκευής όπως είναι το GPS και η Camera. Μόνο με την HTML όμως δεν θα γινόταν να είναι λειτουργική η εφαρμογή. Έτσι με τη βοήθεια της AngularJS η οποία επεκτείνει τη σύνταξη της HTML κάνοντάς τη δυναμική, υλοποιήθηκαν οι λειτουργίες της εφαρμογής. Με τη χρήση των controller και με τη βοήθεια της AngularJS προσφέροντας τα δικά της directives και δένοντας τις HTML σελίδες με τα data της εφαρμογής, η εφαρμογή έγινε λειτουργική. Τέλος, όσον αφορά το user interface, το ionic προσφέρει ένα σύνολο από user interface elements γραμμένα σε HTML και CSS3 που χρησιμοποιούνται για το mobile interaction, μερικά από τα οποία χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής.

Έτσι η mobile εφαρμογή αποτελείται από πολλά αρχεία HTML που είναι για το mobile interaction στο οποίο μέσω της AngularJS έχουμε πρόσβαση στους controller δηλαδή ένα αρχείο Controller.js που περιέχει όλους τους controllers όπου γίνονται όλοι οι υπολογισμοί και οι υλοποιήσεις των αλγορίθμων, και στο αρχείο που έχει όλα τα δεδομένα της εφαρμογής το Services.js. Δηλαδή μέσω της AngularJS γίνεται η σύνδεση των HTML σελίδων με τα data της εφαρμογής. Το αρχείο Services περιέχει όλα τα δεδομένα για τις γραμμές, στάσεις και τα εκδοτήρια εισιτηρίων, εκεί δημιουργούνται όλες οι συναρτήσεις για τη λειτουργία της εφαρμογής.

## 5.2 Προγραμματιστικό περιβάλλον του Ionic

Μια υβριδική εφαρμογή με τη βοήθεια του Ionic μπορεί να δημιουργηθεί τόσο σε Windows όσο και σε Linux λειτουργικό σύστημα. Όσον αφορά την εφαρμογή “VolosBus” δημιουργήθηκε σε Linux περιβάλλον. Τα προγράμματα που χρειάζονται για να αναπτυχθεί μια υβριδική εφαρμογή με ionic είναι τα εξής:

- Java JDK
- Apache Ant
- Android SDK
- NodeJS / NPM
- Apache Cordova
- Ionic Framework

Σημείωση: Στο λειτουργικό σύστημα των Linux υπάρχει αυτοματοποιημένο script το οποίο εγκαθιστά αυτόματα όλα τα παραπάνω.

Εφόσον εγκατασταθούν όλα τα απαραίτητα προγράμματα το μόνο που χρειάζεται είναι ένας επεξεργαστής κειμένου (text editor) για τη δημιουργία κώδικα της εφαρμογής.

Στη συνέχεια αφού έχει γίνει η εγκατάσταση όλων των βασικών προγραμμάτων θα πρέπει να ακολουθηθούν τα εξής βήματα:

- 1) Ανοίγουμε την κονσόλα και γράφουμε την εξής εντολή :

“ionic start onoma\_efarmoghs”

2) Μπορούμε να δούμε την εφαρμογή σε ένα Browser γράφοντας τις εξής εντολές:

i. “cd onoma\_efarmoghs”

ii. “ionic serve”

3) Σε αυτό το σημείο έχουμε δημιουργήσει μία κενή εφαρμογή και το Ionic έχει εισάγει όλα τα απαραίτητα στοιχεία που χρειάζεται η εφαρμογή μας, στη συνέχεια το μόνο που χρειάζεται είναι να προσθέσουμε την πλατφόρμα. Αυτό γίνεται με την εξής εντολή:

“ionic platform add android” (για android κινητά)

“ionic platform add iOS” (για iPhone κινητά)

4) Έπειτα για να δοκιμάσουμε την εφαρμογή, πληκτρολογούμε την εντολή:

“ionic build android”

5) Με αυτή την εντολή άμα επιτύχει το build παράγεται ένα apk το οποίο μπορούμε να εγκαταστήσουμε στην συσκευή μας περνώντας το αρχείο μέσω ενός καλωδίου usb και στην συνέχεια ανοίγοντας το αρχείο από το κινητό μας. Υπάρχει και η δυνατότητα να τρέξουμε την εφαρμογή κατευθείαν στην συσκευή εφόσον είναι συνδεδεμένη με τον υπολογιστή μας και είναι ενεργοποιημένο το usb debugging πληκτρολογώντας την εξής εντολή:

“ionic run android”

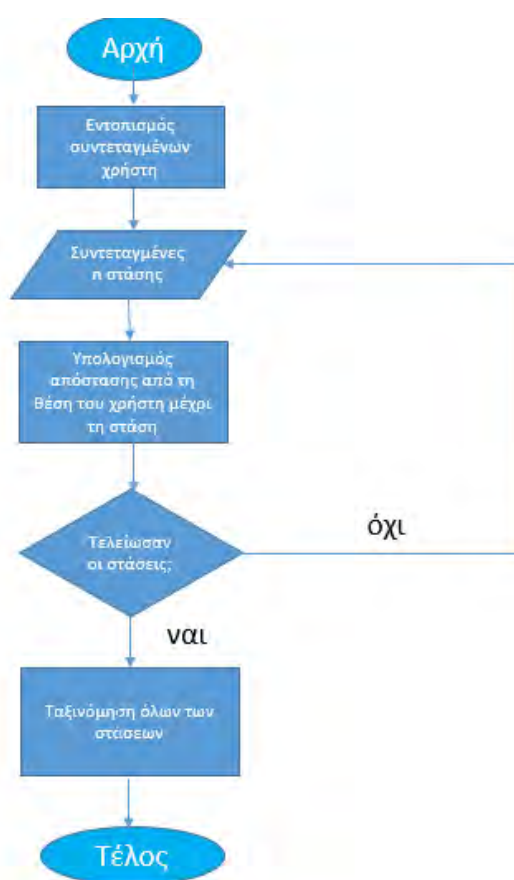
### 5.3 Αναλυτική Παρουσίαση Αλγορίθμων

Στη συνέχεια δεν θα παρουσιαστεί το σύνολο του κώδικα, αλλά θα γίνει παρουσίαση των βασικών και σημαντικών αλγορίθμων της εφαρμογής, δηλαδή θα γίνει εξήγηση πως σχεδιάστηκαν και υλοποιήθηκαν οι αλγόριθμοι με λογικά διαγράμματα ώστε να καλυφθούν οι ανάγκες που είχαν προκύψει σχεδιάζοντας την εφαρμογή.



### 5.3.1 Λειτουργία πλησιέστερες στάσεις

Μια από τις βασικότερες ανάγκες και το κλειδί για τις περισσότερες λειτουργίες της εφαρμογής όπως θα φανεί και στη συνέχεια είναι ο υπολογισμός κοντινότερων στάσεων από την εκάστοτε θέση του χρήστη. Ο αλγόριθμος για την εύρεση πλησιέστερων στάσεων υλοποιήθηκε όπως φαίνεται στο διάγραμμα στην εικόνα 8. Η υλοποίηση του αλγορίθμου έγινε με τη βοήθεια του Cordova όπου με τις κατάλληλες κλίσεις δίνεται η δυνατότητα πρόσβασης στο GPS του κινητού ώστε να ανακτήσουμε τις συντεταγμένες του χρήστη.

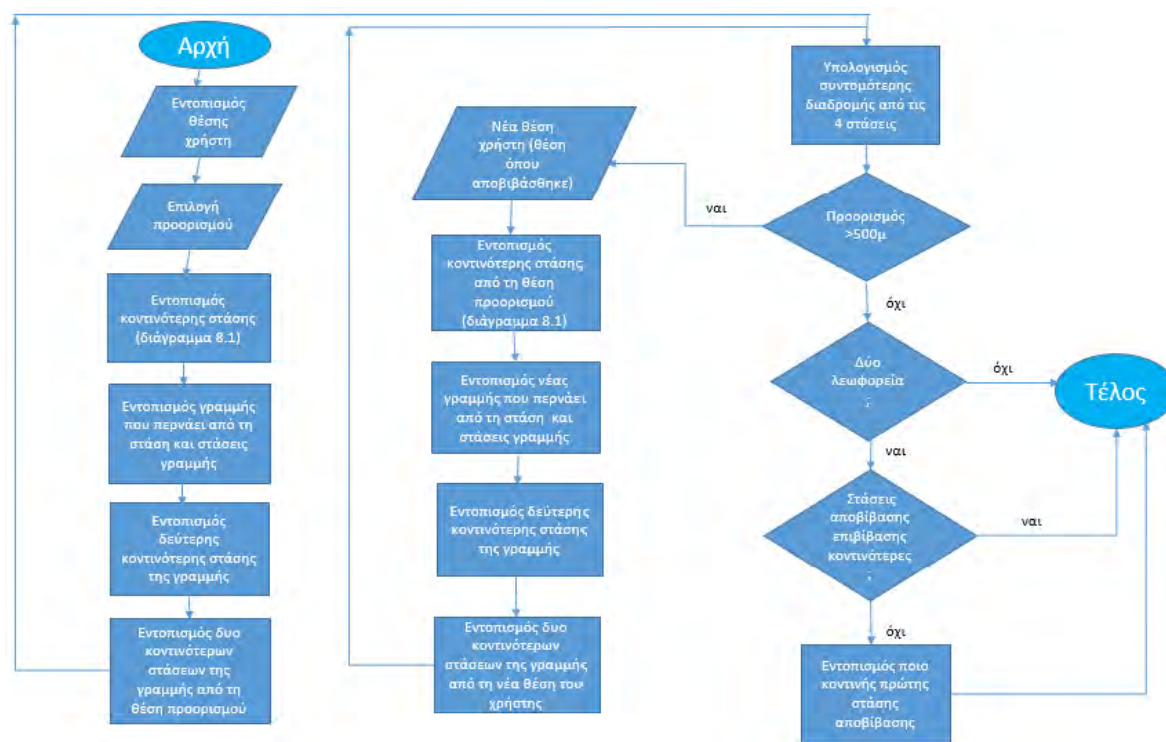


Εικόνα 8

Όπως φαίνεται και στο λογικό διάγραμμα του αλγορίθμου με τη βοήθεια του GPS του κινητού υπολογίζεται η θέση του χρήστη, στη συνέχεια έχοντας τις συντεταγμένες όλων των στάσεων (πληροφορία που λήφθηκε από τους χάρτες της Google) υπολογίζονται οι αποστάσεις της κάθε στάσης με τη θέση του χρήστη. Τέλος γίνεται η ταξινόμηση όλων των αποστάσεων.

### 5.3.2 Αλγόριθμος Βέλτιστης διαδρομής

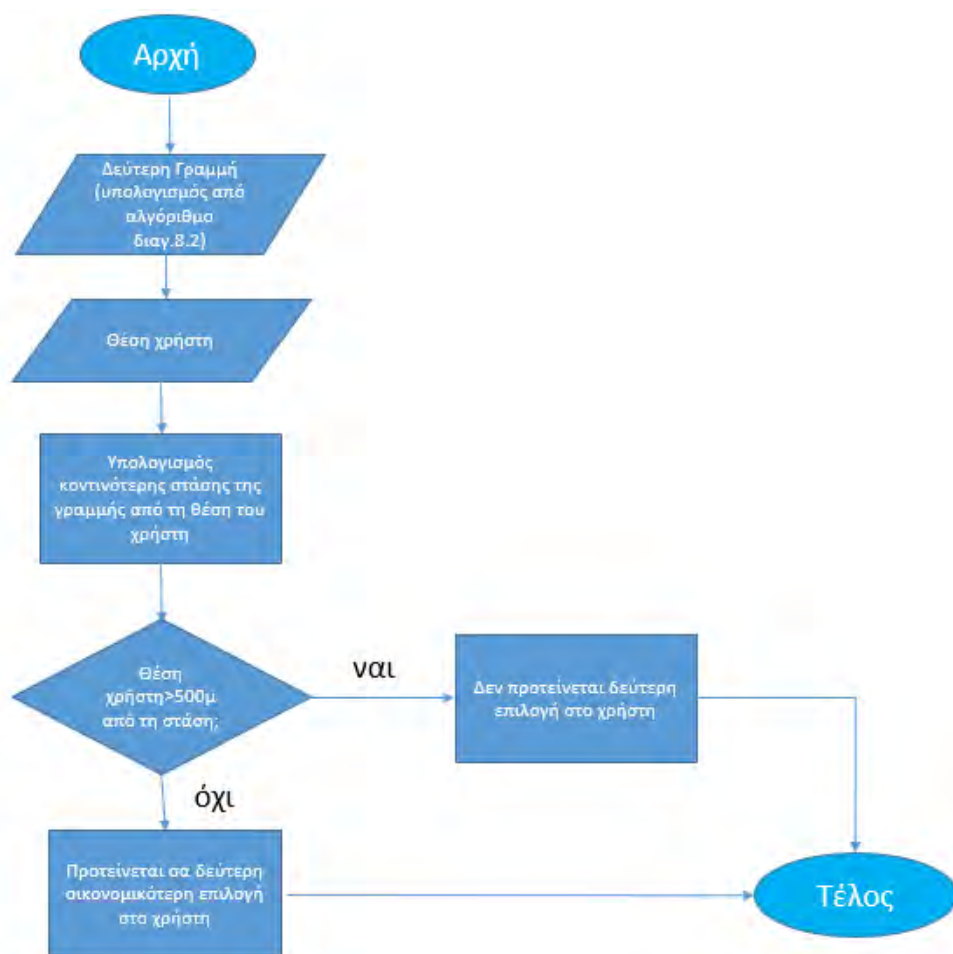
Ο αλγόριθμος Βέλτιστης Διαδρομής δημιουργήθηκε για να προτείνει στο χρήστη τη βέλτιστη και την οικονομικότερη διαδρομή ανάλογα με το προορισμό του. Στην παρακάτω εικόνα απεικονίζεται αναλυτικά το λογικό διάγραμμα ροής των βημάτων για τον υπολογισμό της βέλτιστης διαδρομής. Εικόνα 9.1



Εικόνα 9.1

Για τη δημιουργία του αλγορίθμου της βέλτιστης διαδρομής όπως φαίνεται και από το διάγραμμα έγινε χρήση του αλγορίθμου πλησιέστερης στάσης. Ακόμη, αφού τελειώσει ο υπολογισμός του αλγορίθμου βέλτιστης διαδρομής, αν ο αλγόριθμος έχει αποφασίσει να γίνει η διαδρομή με δύο λεωφορεία θα πρέπει να γίνει εξέταση αν ο χρήστης μπορούσε αντί δυο λεωφορεία να πάρει ένα δηλαδή αν μπορεί να προταθεί στο χρήστη μια πιο οικονομικότερη διαδρομή. Αναλυτικά, θα γίνει έλεγχος εξετάζοντας τη δεύτερη γραμμή, δηλαδή αν υπάρχει κοντινή στάση στην αρχική θέση του χρήστη για το δεύτερο λεωφορείο, δηλαδή εντοπίζει τη κοντινότερη στάση που περνάει η γραμμή του δεύτερου λεωφορείου στην αρχική θέση του χρήστη, αν η απόσταση της αρχικής θέσης του χρήστη με αυτή τη στάση είναι μικρότερη από 500 μέτρα τότε του προτείνει και αυτή σαν δεύτερη επιλογή και πιο οικονομική, αν όχι

προτείνεται μόνο η επιλογή με τα δυο λεωφορεία. Το διάγραμμα της οικονομικότερης διαδρομής φαίνεται στην παρακάτω εικόνα 9.2.

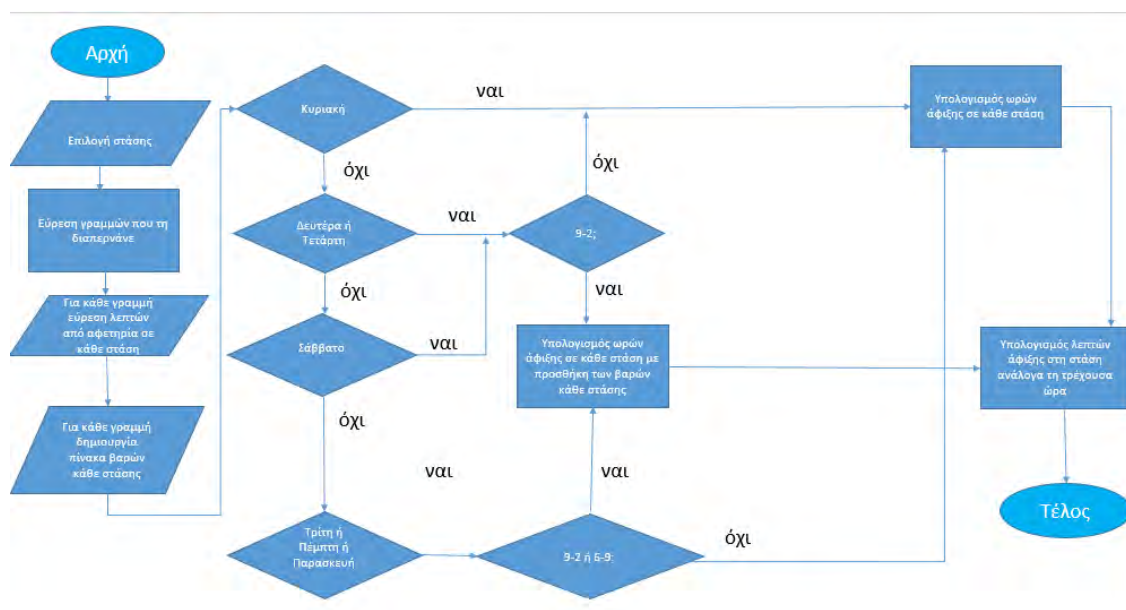


Εικόνα 9.2

### 5.3.3 Αλγόριθμος Αφίξεων σε στάση

Ο αλγόριθμος Αφίξεων σε Στάση υπολογίζει σε πόσα λεπτά αναμένονται οι γραμμές σε κάθε στάση. Για τον υπολογισμό των λεπτών που θα κάνει κάθε γραμμή να φτάσει σε κάθε στάση δόθηκαν κάποιες πληροφορίες από τα ΚΤΕΛ Βόλου. Δηλαδή δόθηκαν για ορισμένες στάσεις τα λεπτά που κάνουν όλες οι γραμμές από την αφετηρία τους προς αυτές. Στη συνέχεια με βάση αυτή τη πληροφορία και τις αποστάσεις κάθε στάσεων στο χάρτη υπολογίστηκαν στο περίπου τα λεπτά που κάνει κάθε λεωφορείο από την αφετηρία του προς όλες τις στάσεις της διαδρομής του. Επίσης δόθηκε σε κάθε στάση κάποιο βάρος δηλαδή ανάλογα το δρόμο που βρίσκεται η στάση, αν είναι κεντρικός, λεωφορειόδρομος ή όχι, δόθηκε αντίστοιχο βάρος έτσι ώστε να υπολογιστεί και η καθυστέρηση στο χρόνο σε ώρες αιχμής,

δηλαδή σε ώρες που έχει περισσότερο κόσμος στις στάσεις και κίνηση στους δρόμους με αποτέλεσμα τα λεπτά που αναμένεται κάθε λεωφορείο σε στάση να αυξάνονται. Παρακάτω στην Εικόνα 10 φαίνεται η λογική του αλγορίθμου που υπολογίζει τις ώρες άφιξης κάθε λεωφορείου στις στάσεις.



Εικόνα 10

#### 5.3.4 Πορεία γραμμών πάνω στο χάρτη

Για την δημιουργία μιας πορείας πάνω στο χάρτη θα πρέπει να τοποθετηθούν πρώτα τα απαραίτητα waypoints. Συγκεκριμένα στο project, για τη δημιουργία της πορείας που διανύει κάθε λεωφορείο, θα πρέπει να τοποθετηθούν πρώτα σαν waypoints όλες οι στάσεις που περνάει το λεωφορείο και έπειτα μέσω μιας συνάρτησης ζωγραφίζεται η πορεία πάνω στο χάρτη ενώνοντας τα waypoints μεταξύ τους. Δυστυχώς όμως το Google Maps API επιτρέπει το κάθε request που είναι για τη δημιουργία μιας πορείας πάνω στο χάρτη να έχει ένα waypoint αρχής, ένα τέλους και έως 8 ενδιάμεσα. Τι γίνεται όμως όταν θέλει κάποιος να δημιουργήσει μια πορεία που να αποτελείται από πολλά waypoints, όπως για παράδειγμα ένα λεωφορείο που περνάει από 20 έως και 30 στάσεις; Η λύση, ήταν να σπάσουμε τη λίστα με τα waypoints σε γκρουπ, κάνοντας πολλαπλά requests στη Google και να ενώσουμε στο τέλος τα αποτελέσματα μεταξύ τους.

## 6 *Επίδειξη της εφαρμογής*

Σε αυτό το κεφάλαιο γίνεται εκτενής παρουσίαση όλης της λειτουργικότητας της εφαρμογής. Παρουσιάζεται ένας πλήρης οδηγός λειτουργίας για το χρήστη ώστε να γίνει κατανοητό πως δουλεύει και τι δυνατότητες προσφέρει η εφαρμογή.

### 6.1 *Γενικά*

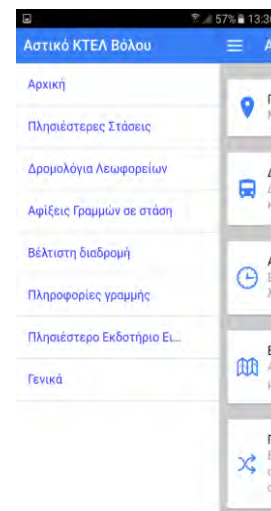
Όπως έχει προαναφερθεί σκοπός της εφαρμογής είναι να διευκολύνει το χρήστη δίνοντας του πληροφορίες για τη μετακίνηση του μέσα στη πόλη του Βόλου με την Αστική Συγκοινωνία. Οι υπηρεσίες που του προσφέρονται λοιπόν είναι (1) απαραίτητες πληροφορίες για το αστικό ΚΤΕΛ (δρομολόγια, στάσεις, ώρες αναχώρησης από την αφετηρία), (2) η δυνατότητα να του προτείνεται η κατάλληλη γραμμή εξυπηρέτησης, μια ή περισσότερες άμα χρειάζονται, όπως επίσης και η πληροφορία σε ποιες στάσεις πρέπει να μεταβεί και σε ποιες πρέπει να αποβιβαστεί. Όταν μια διαδρομή απαιτεί δυο γραμμές, αν η μια έκτων δυο μπορεί να διανυθεί με τα πόδια προτείνονται δυο επιλογές μια οικονομικότερη όπου ο χρήστης μπορεί να επιλέξει να πάρει ένα λεωφορείο και μια ακριβότερη όπου ο χρήστης μπορεί να πάρει δυο λεωφορεία για να φτάσει στο προορισμό του. Επίσης δίνεται η πληροφορία για τα μέτρα που θα πρέπει να διανύσει με τα πόδια μέχρι τις εκάστοτε στάσεις όπως επίσης και τα μέτρα που θα διανύσει με κάθε λεωφορείο, (3) η αναζήτηση των πλησιέστερων στάσεων με βάση τη θέση του χρήστη και η προβολή τους πάνω στο χάρτη, (4) ο χρήστης μπορεί να πληροφορηθεί για το πόσες γραμμές περνάνε από κάθε στάση καθώς επίσης και σε πόσα λεπτά. Δηλαδή, δίνεται πληροφορία σε πόσα λεπτά αναμένονται να περάσουν τα δυο επόμενα λεωφορεία από κάθε γραμμή από την εκάστοτε στάση, (5) η αναζήτηση των πλησιέστερων εκδοτηρίων αφού εντοπίζεται αυτόματα η θέση του χρήστη και η προβολή τους πάνω στο χάρτη. Παρέχονται πληροφορίες που αφορούν τα ωράρια λειτουργίας τους, η διεύθυνση όπως και η απόσταση του χρήστη μέχρι το κάθε εκδοτήριο, (6) η δυνατότητα ενημέρωσης του χρήστη για όλες τις νέες ανακοινώσεις που αφορούν το αστικό ΚΤΕΛ του Βόλου.

## 6.2 Λειτουργία Εκτέλεσης

Μόλις ο χρήστης ανοίξει την εφαρμογή θα του παρουσιαστεί το βασικό και το πλευρικό μενού όπως φαίνεται στις εικόνες 11.1 και 11.2 αντίστοιχα:



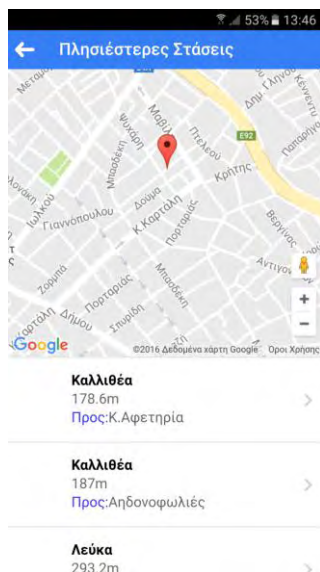
Εικόνα 11.1



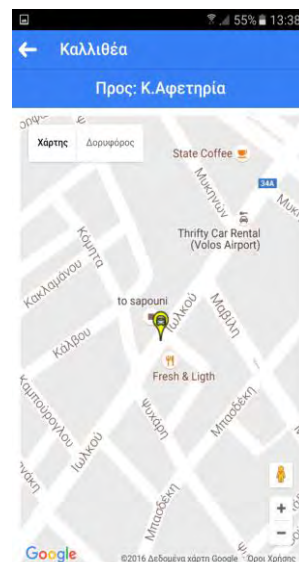
Εικόνα 11.2

Ο χρήστης πατώντας το πρώτο κουμπί «Πλησιέστερες Στάσεις» όπως φαίνεται στην εικόνα 11.1 εντοπίζεται αυτόματα η θέση του μέσω GPS και εμφανίζεται πάνω στο χάρτη όπως φαίνεται στη συνέχεια στην εικόνα 12.1. Παράλληλα εμφανίζεται από κάτω μια λίστα με όλες τις στάσεις ταξινομημένες με βάση την κοντινότερη στη θέση που βρίσκεται ο χρήστης εκείνη τη στιγμή. Επίσης δίνονται πληροφορίες για το όνομα της κάθε στάσης και οι αποστάσεις σε σχέση με τη τοποθεσία του χρήστη.





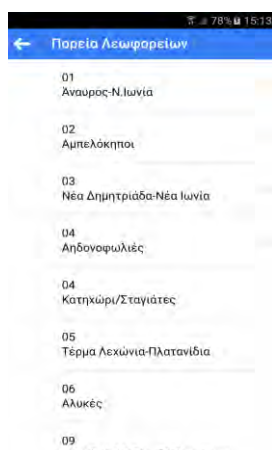
Εικόνα 12.1



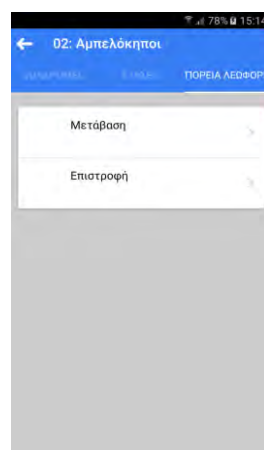
Εικόνα 12.2

Ο χρήστης έχει τη δυνατότητα να δει που βρίσκεται η κάθε στάση που τον ενδιαφέρει στο χάρτη απλώς πατώντας πάνω στη στάση, ένα παράδειγμα παρουσιάζεται στην εικόνα 12.2.

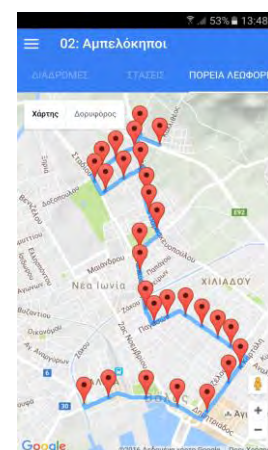
Επιστρέφοντας στο κεντρικό μενού ο χρήστης πατώντας το δεύτερο κουμπί «Δρομολόγια Λεωφορείων» εμφανίζεται μια λίστα με όλες τις γραμμές του αστικού ΚΤΕΛ Βόλου με τίτλο Πορεία Λεωφορείων (εικόνα 13.1).



Εικόνα 13.1



Εικόνα 13.2

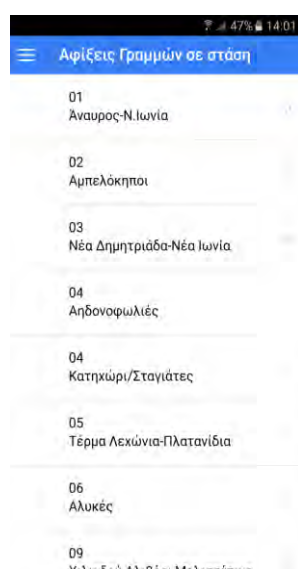


Εικόνα 13.3

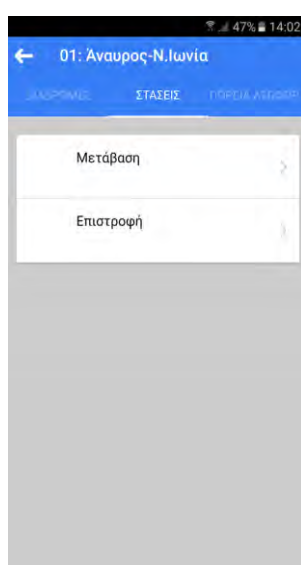
Στη συνέχεια έχει τη δυνατότητα να επιλέξει όποια γραμμή τον ενδιαφέρει και τότε εμφανίζεται να επιλέξει αν θέλει να δει τη πορεία μετάβασης ή επιστροφής που διανύει

η γραμμή που διάλεξε (εικόνα 13.2). Αφού επιλέξει εμφανίζεται πάνω στο χάρτη η πορεία της γραμμής όπως και όλες οι στάσεις που διανύει. Για παράδειγμα άμα επιλέξει τη γραμμή 2 Αμπελόκηποι Μετάβαση θα του εμφανιστεί η πορεία του λεωφορείου όπως φαίνεται στην εικόνα 13.3 με τα waypoints να επισημαίνουν την κάθε στάση που διανύει το λεωφορείο.

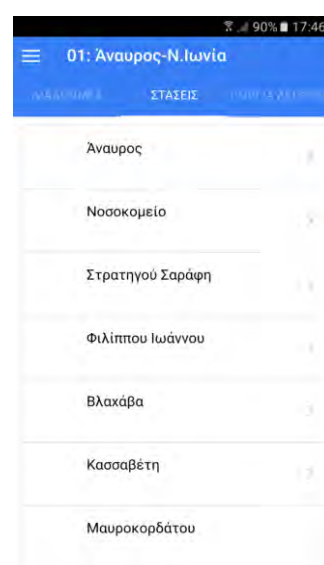
Πατώντας το τρίτο κουμπί «Αφίξεις» από το κεντρικό μενού όπως φαίνεται στην εικόνα 11.1 εμφανίζεται μια λίστα με όλες τις γραμμές του αστικού ΚΤΕΛ Βόλου με τίτλο Αφίξεις Γραμμών σε στάση (εικόνα 14.1).



Εικόνα 14.1



Εικόνα 14.2

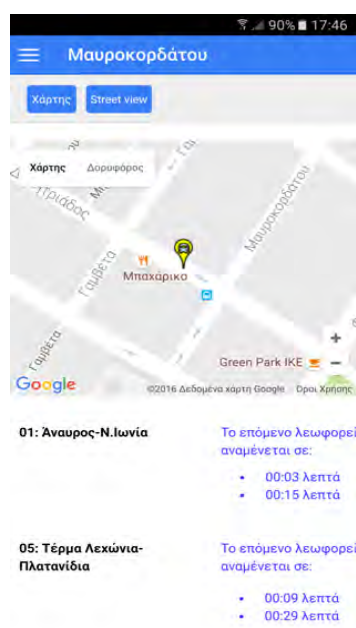


Εικόνα 14.3

Ο χρήστης έχει τη δυνατότητα να επιλέξει όποια γραμμή τον ενδιαφέρει και τότε εμφανίζεται να επιλέξει αν θέλει να δει όλες τις στάσεις μετάβασης ή επιστροφής της γραμμής που επέλεξε όπως φαίνεται στην εικόνα 14.2. Αφού επιλέξει, εμφανίζεται μια λίστα με όλες τις στάσεις που διανύει η εκάστοτε γραμμή που επέλεξε (εικόνα 14.3). Εκεί έχει τη δυνατότητα να επιλέξει όποια στάση θέλει έτσι ώστε να πληροφορηθεί σε πόσα λεπτά θα περάσει από εκεί η γραμμή που τον ενδιαφέρει. Έτσι μόλις επιλέξει τη στάση που προτιμάει εμφανίζεται ένας χάρτης που δείχνει το σημείο που βρίσκεται η στάση καθώς και επίσης όλες οι γραμμές που περνάνε από τη συγκεκριμένη στάση. Στη συνέχεια ο χρήστης μπορεί να πληροφορηθεί σε πόσα λεπτά θα περάσουν από εκείνη τη στάση τα δυο επόμενα λεωφορεία από τη γραμμή που τον ενδιαφέρει. Για



παράδειγμα άμα πατήσει πάνω στη στάση Μαυροκορδάτου θα του εμφανιστεί ο χάρτης δείχνοντας το σημείο που βρίσκεται η στάση και από κάτω ποια λεωφορεία περνάνε από εκεί καθώς επίσης και σε πόση ώρα όπως φαίνεται στην εικόνα 14.4.



Εικόνα 14.4

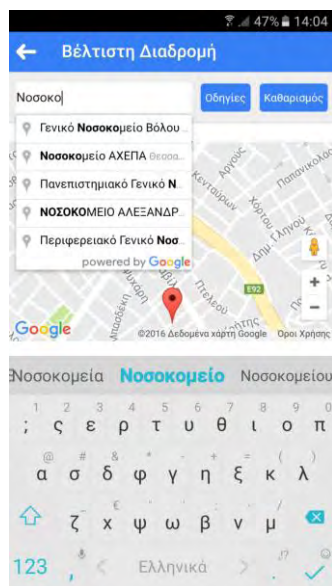


Εικόνα 14.5

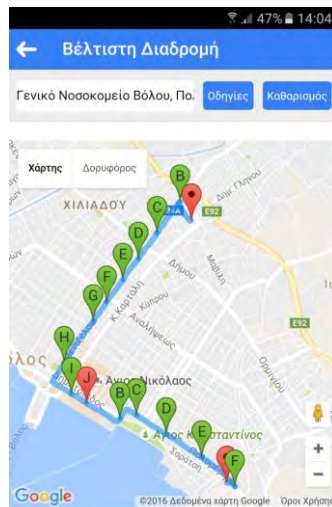
Επίσης παρέχεται στο χρήστη η δυνατότητα να δει και τη κάθε στάση μέσω Street View (εικόνα 14.5) πατώντας το κουμπί «**Street View**» και αν θέλει να ξαναδεί τη στάση πάνω στο χάρτη, αρκεί να επιλέξει το κουμπί «**Χάρτης**».

Στο σημείο αυτό να αναφέρουμε ότι στα λεπτά έχει ληφθεί υπόψη και οι ώρες αιχμής. Δηλαδή ώρες που υπάρχει αρκετή κίνηση στο δρόμο λόγω άλλων οχημάτων ή λόγω υπερβολικού κόσμου σε κάθε στάση(ώρες που είναι ανοιχτά τα μαγαζιά ή κόσμος που επιστρέφει από δουλειές) με αποτέλεσμα τα λεωφορεία να κάνουν περισσότερο χρόνο από τη μια στάση στην άλλη.

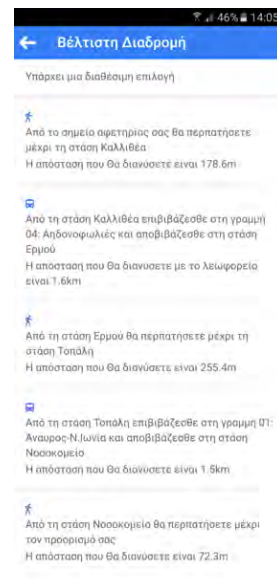
Επιστρέφοντας στο βασικό μενού ο χρήστης έχει τη δυνατότητα να πληροφορηθεί ποια ή ποιες γραμμές να επιλέξει με βάση το προορισμό του. Δηλαδή αρκεί να πατήσει το κουμπί «**Βέλτιστη Διαδρομή**». Αυτόματα εντοπίζεται η θέση του χρήστη μέσω GPS παρουσιάζοντας τη πάνω στο χάρτη, η εφαρμογή αναμένει να θέσει ο χρήστης οποιοδήποτε προορισμό θέλει (διεύθυνση ή κάποιο σημείο ενδιαφέροντος) όπως φαίνεται στην εικόνα 15.1. Αφού θέσει τον προορισμό που επιθυμεί, εμφανίζεται στο χάρτη όλη η διαδρομή που πρέπει να διανύσει μέχρι το προορισμό του (εικόνα 15.2).



Εικόνα 15.1



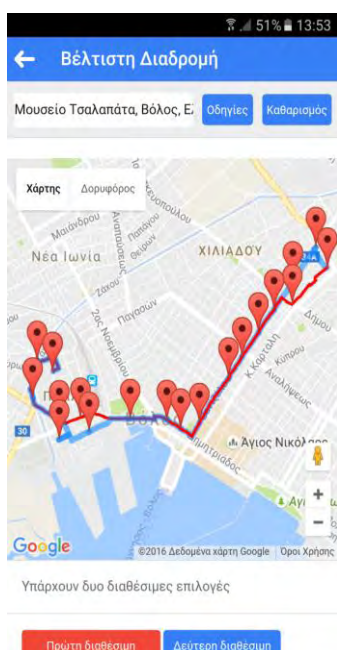
Εικόνα 15.2



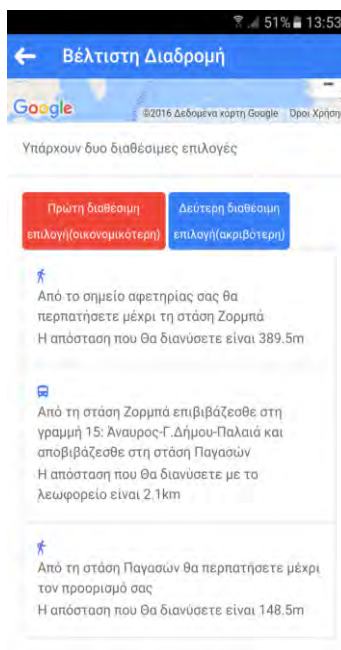
Εικόνα 15.3

Ποιο αναλυτικά εμφανίζεται στο χάρτη το σημείο που βρίσκεται ο χρήστης, η στάση που πρέπει να μεταβεί, όλη η διαδρομή που θα διανύσει με το λεωφορείο ή τα λεωφορεία που πρέπει να επιλέξει, μαζί με τις στάσεις που θα περάσει καθώς επίσης και τη στάση που πρέπει να αποβιβαστεί και το σημείο που πρέπει να κατευθυνθεί με τα πόδια για να φτάσει στον τελικό προορισμό του. Στη συνέχεια ο χρήστης έχει τη δυνατότητα να διαβάσει όλη αυτή τη πληροφορία πιο αναλυτικά με λόγια όπως παρουσιάζεται στην εικόνα 15.3 πατώντας το κουμπί «Οδηγίες». Η εφαρμογή ενημερώνει το χρήστη σε ποια στάση θα πρέπει να μεταβεί και σε πόσα μέτρα, στη συνέχεια τον πληροφορεί σε ποια γραμμή πρέπει να επιβιβαστεί και σε ποια στάση πρέπει να αποβιβαστεί καθώς και πόσα μέτρα έχει διανύσει με το λεωφορείο. Έπειτα αν χρειάζεται να επιβιβαστεί και σε δεύτερη γραμμή τον ενημερώνει σε ποια στάση να μεταβεί και σε πόσα μέτρα βρίσκεται αυτή η στάση από εκεί που κατέβηκε όπως επίσης και σε ποια γραμμή πρέπει να επιβιβαστεί. Τέλος, τον ενημερώνει σε ποια στάση πρέπει να αποβιβαστεί και σε πόσα μέτρα από εκεί βρίσκεται ο προορισμός που έχει επιλέξει να πάει.

Αν ο προορισμός που επιλέξει ο χρήστης χρειάζεται δυο γραμμές αλλά η διαδρομή μπορεί να καλυφθεί και με ένα λεωφορείο χωρίς να περπατήσει πολύ ο χρήστης τότε η εφαρμογή προτείνει δυο επιλογές, μια οικονομική και μια πιο ακριβή καθώς θα χρειαστεί να πάρει 2 γραμμές άρα και 2 εισιτήρια. Στην οικονομική διαδρομή τον ενημερώνει να μεταβεί στη πιο κοντινή του στάση της δεύτερης γραμμής και να πάρει το λεωφορείο που του προτείνει μέχρι το προορισμό του. Στη πιο ακριβή επιλογή του επιδεικνύει να μεταβεί στην πιο κοντινή στάση με βάση την θέση του και να πάρει τη πρώτη γραμμή που του προτείνει, στη συνέχεια να μεταβεί σε δεύτερη στάση ώστε να πάρει το δεύτερο προτεινόμενο λεωφορείο. Η επιλογή είναι στο χρήστη. Ένα παράδειγμα εμφανίζεται στις εικόνες 15.4 15.5 και 15.6 όπου ο χρήστης έχει τη δυνατότητα να επιλέξει να πάρει ένα λεωφορείο και να ακολουθήσει τη κόκκινη γραμμή που του παρουσιάζεται στο χάρτη ή να επιλέξει να πάρει δυο λεωφορεία και να ακολουθήσει τη μπλε γραμμή.



Εικόνα 15.4



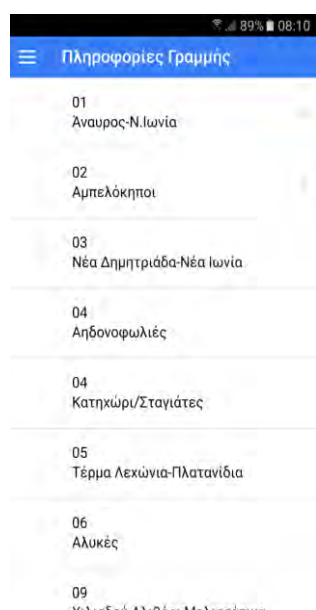
Εικόνα 15.5



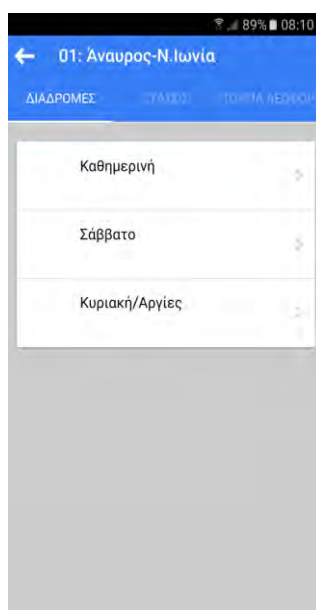
Εικόνα 15.6

Ο χρήστης έχει τη δυνατότητα πατώντας το κουμπί «Καθαρισμός» όπως φαίνεται στην εικόνα 15.1 και 15.2, να καθαρίσει το χάρτη και στη συνέχεια να επιλέξει κάποιον άλλο νέο προορισμό.

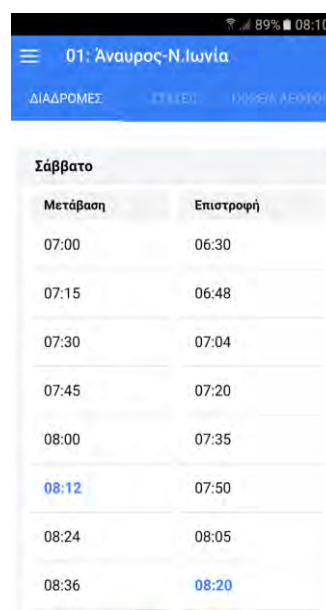
Επόμενη δυνατότητα που δίνεται στο χρήστη είναι ότι επιλέγοντας το κουμπί «Γραμμές» από το βασικό μενού, εμφανίζεται στο χρήστη μια λίστα με όλες τις γραμμές του αστικού ΚΤΕΛ Βόλου και με τίτλο Πληροφορίες Γραμμής όπως φαίνεται στην Εικόνα 16.1.



Εικόνα 16.1



Εικόνα 16.2



Εικόνα 16.3

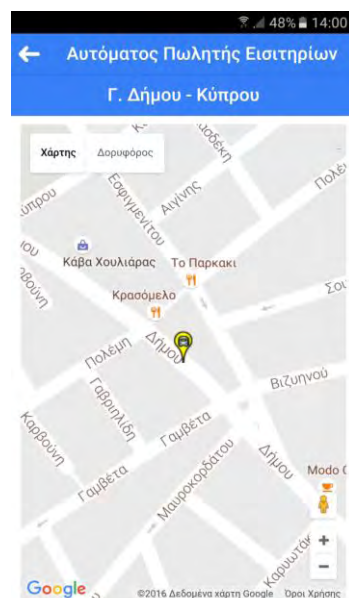
Ο χρήστης επιλέγει όποια γραμμή τον ενδιαφέρει και του εμφανίζεται να επιλέξει για ποια μέρα της εβδομάδας θέλει να πληροφορηθεί για τις ώρες αναχωρήσεως των λεωφορείων της γραμμής από την αφετηρία τους (Εικόνα 16.2) . Στη συνέχεια αφού επιλέξει τη μέρα, Καθημερινή Σάββατο ή Κυριακή εμφανίζονται οι ώρες αναχωρήσεως των λεωφορείων για τη συγκεκριμένη γραμμή μετάβασης και επιστροφής καθώς και με χρωματική ένδειξη το επόμενο λεωφορείο που αναμένεται να αναχωρήσει από την αφετηρία με βάση τη τρέχουσα ώρα όπως φαίνεται στην Εικόνα 16.3.

Επίσης για διευκόλυνση του χρήστη δίνεται η δυνατότητα να μπορεί να πληροφορηθεί για μια συγκεκριμένη γραμμή που διάλεξε τόσο για τις στάσεις και τις ώρες αφίξεως των γραμμών σε αυτές, όσο και για τις ώρες αναχωρήσεως από την αφετηρία και την πορεία των δρομολογίων χωρίς να βγει στο κεντρικό μενού μέσω του tab που φαίνεται στις παραπάνω εικόνες 16.2 και 16.3.

Επόμενη λειτουργία εκτέλεσης είναι το «**Κοντινότερο Εκδοτήριο Εισιτηρίων**» . Με το πάτημα αυτού του κουμπιού υπολογίζεται αυτόματα μέσω GPS η θέση του χρήστη και εμφανίζεται πάνω στο χάρτη. Παράλληλα εμφανίζεται μια λίστα με όλα τα εκδοτήρια και αυτόματους πωλητές εισιτηρίων ταξινομημένα με βάση την κοντινότερη στη θέση του χρήστη όπως φαίνεται στην Εικόνα 17.1. Επίσης δίνονται πληροφορίες για τη διεύθυνση των εκδοτηρίων και τις αποστάσεις σε σχέση με τη τοποθεσία του χρήστη καθώς και για τα ωράρια λειτουργίας τους. Έτσι ώστε αν κάποιος εκδοτήριο εκείνη την ώρα είναι κλειστό να μεταβεί στο επόμενο κοντινότερο που θα είναι ανοιχτό.



Εικόνα 17.1

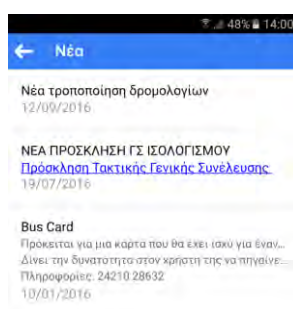


Εικόνα 17.2

Ο χρήστης έχει τη δυνατότητα να δει που βρίσκεται το κάθε εκδοτήριο εισιτηρίων που τον ενδιαφέρει στο χάρτη απλώς πατώντας πάνω στο εκδοτήριο όπως παρουσιάζεται στην Εικόνα 17.2.

Τέλος, επιστρέφοντας στο αρχικό μενού ο χρήστης πατώντας το τελευταίο κουμπί «**Νέα**» έχει τη δυνατότητα να πληροφορηθεί για όλες τις πρόσφατες ανακοινώσεις του Αστικού ΚΤΕΛ Βόλου όπως φαίνεται στην Εικόνα 18.





*Εικόνα 18*

## 7 Συμπεράσματα

Σε αυτό το τελευταίο κεφάλαιο περιγράφονται τα αποτελέσματα που είχε η ανάπτυξη της εφαρμογής, οι δυσκολίες που αντιμετωπίστηκαν κατά τη πορεία της δημιουργίας της και τέλος αναφέρονται πιθανές μελλοντικές επεκτάσεις της .

### 7.1 Αποτελέσματα

Με το πέρας της μεταπτυχιακής, οι στόχοι της δημιουργίας μια υβριδικής εφαρμογής που ο χρήστης μπορεί να κατεβάσει και να εγκαταστήσει στο κινητό του ανεξαρτήτως λειτουργικού συστήματος, και η ανάπτυξη μιας εφαρμογής που θα ήταν απλή στη χρήση αλλά θα πρόσφερε σημαντικές πληροφορίες για τη διευκόλυνση του χρήστη μέσα στη πόλη του Βόλου με την αστική συγκοινωνία επιτεύχθηκαν.

Κατά τη διάρκεια της εργασίας μελετήθηκε σε βάθος η ανάπτυξη υβριδικών εφαρμογών. Έχοντας πλέον περισσότερη εμπειρία στον προγραμματισμό και στην ανάπτυξη υβριδικών εφαρμογών μπορώ να αναφέρω σα συμπέρασμα ότι είναι πολύ όμορφο και ενδιαφέρον αντικείμενο να ασχοληθεί κανείς.

### 7.2 Δυσκολίες

Κατά τη διάρκεια της εργασίας αντιμετώπισα μερικές δυσκολίες. Η πρώτη και σημαντικότερη ήταν η εξοικείωση με το Ionic και γενικά με την όλη διαδικασία δημιουργίας υβριδικών εφαρμογών, μετά από μελέτη και αφοσίωση οι δυσκολίες αυτές ξεπεράστηκαν γρήγορα. Επόμενη δυσκολία που αντιμετώπισα ήταν στη δημιουργία ΠΟΡΕΙΑ ΛΕΩΦΟΡΕΙΩΝ κατά τη διαδικασία απεικόνισης της πορείας και όλων των στάσεων πάνω στο χάρτη. Η δυσκολία ήταν ότι η Google σου προσφέρει να βάλεις μόνο ως 8 waypoints πάνω στο χάρτη, κάτι που καθιστούσε αδύνατον τη δημιουργία πορείας λεωφορείων που περνάνε από 20 και 30 στάσεις. Συνεπώς έπρεπε να σκεφτώ ένα τρόπο για την υλοποίηση της συγκεκριμένης πορείας χωρίς να παρεκκλίνει από τη σωστή πορεία των λεωφορείων.

### 7.3 Μελλοντικές επεκτάσεις

Όλες οι εφαρμογές συνήθως αναβαθμίζονται και ανανεώνονται καθώς προκύπτουν νέες ανάγκες με αποτέλεσμα να είναι πολύ σπάνιο μια εφαρμογή να μείνει στην αρχική της έκδοση. Αυτό αποτελεί στόχο και για την παρούσα εφαρμογή.

Η επόμενη έκδοση θα μπορούσε να περιέχει:

1. Ένα πλήρη οδηγό αναψυχής και διαμονής στο Βόλο. Ο χρήστης θα έχει τη δυνατότητα να αναζητά κοντινά ξενοδοχεία, καφετέριες, μουσεία και ότι άλλο τον ενδιαφέρει μέσω της θέσης του.
2. Πληροφορίες για τις πιάτσες των ταξί. Θα δίνεται η δυνατότητα στο χρήστη να ενημερώνεται για τις κοντινότερες πιάτσες ταξί με βάση τη θέση του. Επίσης ενδεχομένως σε κάποια συνεργασία με τα ταξί του Βόλου θα του δίνεται η δυνατότητα να καλέσει μέσω της εφαρμογής ταξί.
3. Επέκταση της εφαρμογής για τη μετακίνηση του χρήστη εκτός Βόλου. Για παράδειγμα αν ο χρήστης θέλει να ταξιδέψει εκτός Βόλου να του προτείνεται ποιο Υπεραστικό λεωφορείο ή ποιο τραίνο πρέπει να πάρει και ποιες ώρες. Επίσης σε συνεργασία με το υπεραστικό κτελ και τη τραινωση να έχει τη δυνατότητα μέσω εφαρμογής να κάνει κράτηση εισιτηρίου.
4. Η χρήση κεντρικού server για την ανανέωση των δεδομένων της εφαρμογής με την χρήση του διαδικτύου προκειμένου να αποφεύγεται η δημιουργία νέας έκδοσης στην περίπτωση αλλαγής των ωρών και των δρομολογίων.



## 8 Βιβλιογραφία

- [1] Android: [goo.gl/JAGWfa](https://goo.gl/JAGWfa)
- [2] iOS: <https://en.wikipedia.org/wiki/IOS>
- [3] Windows phone: [https://en.wikipedia.org/wiki/Windows\\_Phone](https://en.wikipedia.org/wiki/Windows_Phone)
- [4] Ionic: <https://ionicframework.com/>
- [5] Cordova: <https://cordova.apache.org/>
- [6] AngularJS: <https://angularjs.org/>
- [7] Αστικό Κτελ Βόλου: <http://astikovoulou.gr/>
- [8] O.A.Σ.Θ. bus App: [goo.gl/RyoiKw](https://goo.gl/RyoiKw)
- [9] O.A.S.A. Telematics App : [goo.gl/Jt7mBx](https://goo.gl/Jt7mBx)
- [10] HTML: <https://www.w3schools.com/html/>
- [11] CSS: <https://www.w3schools.com/css/>
- [12] JavaScript: <https://www.w3schools.com/js/default.asp>
- [13] Sencha Touch: <https://goo.gl/NcnVHV>
- [14] Onsen UI: <https://onsen.io/>

- [15] Kendo UI: <http://www.telerik.com/kendo-ui>
- [16] jQuery Mobile: <https://jquerymobile.com/>
- [17] Adobe PhoneGap: <https://build.phonegap.com/>
- [18] Google Maps Android API: <https://goo.gl/hN3OFI>

